S01P1590US00

日本国特許庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出願年月日
Date of Application: ２０００年１０月２０日

出願番号
Application Number: 特願２０００－３２０８０５

出願人
Applicant(s): ソニー株式会社

CERTIFIED COPY OF
PRIORITY DOCUMENT

２００１年 ８月３１日

特許庁長官
Commissioner,
Japan Patent Office

及川耕造

【書類名】　特許願

【整理番号】　0000647705

【提出日】　平成12年10月20日

【あて先】　特許庁長官殿

【国際特許分類】　G06F 13/00

【発明者】

【住所又は居所】　東京都品川区北品川６丁目７番３５号　ソニー株式会社内

【氏名】　白井　太三

【発明者】

【住所又は居所】　東京都品川区北品川６丁目７番３５号　ソニー株式会社内

【氏名】　石橋　義人

【発明者】

【住所又は居所】　東京都品川区北品川６丁目７番３５号　ソニー株式会社内

【氏名】　吉野　賢治

【発明者】

【住所又は居所】　東京都品川区北品川６丁目７番３５号　ソニー株式会社内

【氏名】　秋下　徹

【特許出願人】

【識別番号】　000002185

【氏名又は名称】　ソニー株式会社

【代表者】　出井　伸之

【代理人】

【識別番号】　100101801

【弁理士】

【氏名又は名称】　山田　英治

【電話番号】　03-5541-7577

【選任した代理人】

【識別番号】　100093241

【弁理士】

【氏名又は名称】　宮田　正昭

【電話番号】　03-5541-7577

【選任した代理人】

【識別番号】　100086531

【弁理士】

【氏名又は名称】　澤田　俊夫

【電話番号】　03-5541-7577

【手数料の表示】

【予納台帳番号】　062721

【納付金額】　21,000円

【提出物件の目録】

【物件名】　明細書　1

【物件名】　図面　1

【物件名】　要約書　1

【包括委任状番号】　9904833

【プルーフの要否】　要

【書類名】　明細書

【発明の名称】　データ記憶装置、およびデータ記録方法、データ再生方法、並びにプログラム提供媒体

【特許請求の範囲】

【請求項１】

各々が予め定められたデータ容量を持つ複数のセクタを１ブロックとしたブロックを複数有するデータ格納領域と、暗号処理手段とを有するデータ記憶装置において、

前記暗号処理手段は、

前記データ記憶装置とデータ通信可能なデバイスから、前記データ格納領域に格納するデータの暗号処理鍵として、前記複数のセクタ各々の暗号処理用鍵または復号処理用鍵として対応づけられた複数の鍵からなる鍵セットを受領し、該受領した鍵セットに対してＣＢＣモードでの暗号処理を実行して、前記デバイスに送信する処理を実行する構成を有することを特徴とするデータ記憶装置。

【請求項２】

前記暗号処理手段は、

前記鍵セットに対するＣＢＣモードでの暗号処理を、前記データ記憶装置に固有の保存鍵Ｋｓｔｏを適用して実行して前記データ格納領域に格納するデータのヘッダ情報としての鍵データを生成する構成を有することを特徴とする請求項１に記載のデータ記憶装置。

【請求項３】

前記データ記憶装置は、

前記データ記憶装置とデータ通信可能な前記デバイスとの相互認証処理を実行する構成を有し、

前記デバイスから受領する鍵セットは、前記相互認証時に生成したセッション鍵を用いたＣＢＣモードによる暗号化を行なったセッション鍵適用ＣＢＣモード処理鍵セットであり、

前記暗号処理手段は、

前記セッション鍵適用ＣＢＣモード処理鍵セットを前記セッション鍵を用いた

ＣＢＣモードで復号し、

復号した鍵セットを前記データ記憶装置に固有の保存鍵Ｋｓｔｏに基づいてＣＢＣモードで暗号処理を実行して、保存鍵適用ＣＢＣモード処理鍵セットを生成して前記デバイスにヘッダ情報構成データとして送信する処理を実行する構成を有することを特徴とする請求項１に記載のデータ記憶装置。

【請求項４】

前記データ記憶装置は、

前記データ記憶装置とデータ通信可能な前記デバイスとの相互認証処理を実行する構成を有し、

前記デバイスから受領する鍵セットは、前記データ格納領域に格納するデータのヘッダ情報として格納された鍵セットであって、前記データ記憶装置に固有の保存鍵Ｋｓｔｏに基づいてＣＢＣモードで暗号処理を実行した保存鍵適用ＣＢＣモード処理鍵セットであり、

前記暗号処理手段は、

前記保存鍵適用ＣＢＣモード処理鍵セットを保存鍵を用いたＣＢＣモードで復号し、

復号した鍵セットを前記相互認証時に生成したセッション鍵に基づいてＣＢＣモードで暗号処理を実行して、セッション鍵適用ＣＢＣモード処理鍵セットを生成して前記デバイスにデータ復号用鍵情報構成データとして送信する処理を実行する構成を有することを特徴とする請求項１に記載のデータ記憶装置。

【請求項５】

前記暗号処理手段は、

前記データ記憶装置とデータ通信可能なデバイスから、前記データ格納領域に格納するデータの暗号処理鍵として、前記複数のセクタ各々の暗号処理用鍵または復号処理用鍵として対応づけられた複数の鍵からなる鍵セットと、

前記複数のセクタの少なくともいずれかに格納されるデータの改竄チェック値（ＩＣＶ）生成用の鍵Ｋｉｃｖと、

を受領し、該受領した鍵セットに対してＣＢＣモードでの暗号処理を実行して、前記デバイスに送信する処理を実行する構成を有することを特徴とする請求項

１に記載のデータ記憶装置。

【請求項６】

各々が予め定められたデータ容量を持つ複数のセクタを１ブロックとしたブロックを複数有するデータ格納領域と、暗号処理手段とを有するデータ記憶装置と、

該データ記憶装置にデータを送信してデータの格納処理を実行するデータ記録装置とを有するデータ処理装置におけるデータ記録方法において、

前記データ記憶装置および前記データ記録装置間において相互認証処理を実行するステップと、

前記相互認証の成立を条件として、前記データ記録装置において、前記複数のセクタ各々の格納データの暗号処理に適用する複数の暗号鍵からなる暗号鍵セットを前記相互認証時に生成したセッション鍵を用いたＣＢＣモードによって暗号化して、セッション鍵適用ＣＢＣモード処理鍵セットを生成して、前記データ記憶装置に対して送信するステップと、

前記データ記憶装置において、前記セッション鍵適用ＣＢＣモード処理鍵セットをセッション鍵を適用したＣＢＣモードで復号するステップと、

復号した鍵セットを前記データ記憶装置に固有の保存鍵Ｋｓｔｏに基づいてＣＢＣモードで暗号化して、保存鍵適用ＣＢＣモード処理鍵セットを生成して、前記データ記録装置に送信するステップと、

前記データ記録装置において、前記データ記憶装置に格納するデータに対応するヘッダ情報として、受信した前記保存鍵適用ＣＢＣモード処理鍵セットデータを構成要素として含むヘッダ情報を生成するステップと、

を有することを特徴とするデータ記録方法。

【請求項７】

各々が予め定められたデータ容量を持つ複数のセクタを１ブロックとしたブロックを複数有するデータ格納領域と、暗号処理手段とを有するデータ記憶装置と、

該データ記憶装置からデータを受信してデータの再生処理を実行するデータ再生装置とを有するデータ処理装置におけるデータ再生方法において、

前記データ記憶装置および前記データ再生装置間において相互認証処理を実行するステップと、

前記相互認証の成立を条件として、前記データ格納領域に格納されたデータのヘッダ情報に含まれる鍵セットであって、前記データ記憶装置に固有の保存鍵Ｋｓｔｏに基づいてＣＢＣモードで暗号処理を実行した保存鍵適用ＣＢＣモード処理鍵セットを前記データ記録装置から前記データ記憶装置に送信するステップと、

前記データ記憶装置において、

前記保存鍵適用ＣＢＣモード処理鍵セットを保存鍵を用いたＣＢＣモードで復号するステップと、

復号した鍵セットを前記相互認証時に生成したセッション鍵に基づいてＣＢＣモードで暗号処理を実行して、セッション鍵適用ＣＢＣモード処理鍵セットを生成して前記データ再生装置に送信するステップと、

前記データ再生装置において、前記セッション鍵適用ＣＢＣモード処理鍵セットをセッション鍵を適用したＣＢＣモードにより復号して、前記データ格納領域の各セクタに格納された暗号化セクタデータの復号用の鍵セットを取得するステップと、

を有することを特徴とするデータ再生方法。

【請求項８】

各々が予め定められたデータ容量を持つ複数のセクタを１ブロックとしたブロックを複数有するデータ格納領域と、暗号処理手段とを有するデータ記憶装置と、

該データ記憶装置にデータを送信してデータの格納処理を実行するデータ記録装置とを有するデータ処理装置におけるデータ記録処理をコンピュータ・システム上で実行せしめるコンピュータ・プログラムを提供するプログラム提供媒体であって、前記コンピュータ・プログラムは、

前記データ記憶装置および前記データ記録装置間において相互認証処理を実行するステップと、

前記相互認証の成立を条件として、前記データ記録装置において、前記複数の

セクタ各々の格納データの暗号処理に適用する複数の暗号鍵からなる暗号鍵セットを前記相互認証時に生成したセッション鍵を用いたＣＢＣモードによって暗号化して、セッション鍵適用ＣＢＣモード処理鍵セットを生成して、前記データ記憶装置に対して送信するステップと、

前記データ記憶装置において、前記セッション鍵適用ＣＢＣモード処理鍵セットをセッション鍵を適用したＣＢＣモードで復号するステップと、

復号した鍵セットを前記データ記憶装置に固有の保存鍵Ｋｓｔｏに基づいてＣＢＣモードで暗号化して、保存鍵適用ＣＢＣモード処理鍵セットを生成して、前記データ記録装置に送信するステップと、

前記データ記録装置において、前記データ記憶装置に格納するデータに対応するヘッダ情報として、受信した前記保存鍵適用ＣＢＣモード処理鍵セットデータを構成要素として含むヘッダ情報を生成するステップと、

を有することを特徴とするプログラム提供媒体。

【請求項９】

各々が予め定められたデータ容量を持つ複数のセクタを１ブロックとしたブロックを複数有するデータ格納領域と、暗号処理手段とを有するデータ記憶装置と、

該データ記憶装置からデータを受信してデータの再生処理を実行するデータ再生装置とを有するデータ処理装置におけるデータ再生処理をコンピュータ・システム上で実行せしめるコンピュータ・プログラムを提供するプログラム提供媒体であって、前記コンピュータ・プログラムは、

前記データ記憶装置および前記データ再生装置間において相互認証処理を実行するステップと、

前記相互認証の成立を条件として、前記データ格納領域に格納されたデータのヘッダ情報に含まれる鍵セットであって、前記データ記憶装置に固有の保存鍵Ｋｓｔｏに基づいてＣＢＣモードで暗号処理を実行した保存鍵適用ＣＢＣモード処理鍵セットを前記データ記録装置から前記データ記憶装置に送信するステップと、

前記データ記憶装置において、

前記保存鍵適用ＣＢＣモード処理鍵セットを保存鍵を用いたＣＢＣモードで復号するステップと、

復号した鍵セットを前記相互認証時に生成したセッション鍵に基づいてＣＢＣモードで暗号処理を実行して、セッション鍵適用ＣＢＣモード処理鍵セットを生成して前記データ再生装置に送信するステップと、

前記データ再生装置において、前記セッション鍵適用ＣＢＣモード処理鍵セットをセッション鍵を適用したＣＢＣモードにより復号して、前記データ格納領域の各セクタに格納された暗号化セクタデータの復号用の鍵セットを取得するステップと、

を有することを特徴とするプログラム提供媒体。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】

本発明は、データ記憶装置、およびデータ記録方法、データ再生方法、並びにプログラム提供媒体に関する。特に、記憶装置に格納されるコンテンツを高度なセキュリティ管理のもとに保護することを可能とするデータ記憶装置、およびデータ記録方法、データ再生方法、並びにプログラム提供媒体に関する。

【０００２】

【従来の技術】

近年のインターネットの急激な普及、さらにモバイル型の小型再生器、ゲーム器等の普及に伴い、音楽データ、ゲームプログラム、画像データ等、様々なソフトデータ（以下、これらをコンテンツ（Content）と呼ぶ）の、インターネット等のネットワーク、あるいは、ＤＶＤ、ＣＤ、メモリカード等の記憶媒体を介した流通が急増している。これらの流通コンテンツは、ユーザの所有するＰＣ（Personal Computer）、再生専用器、あるいはゲーム機器においてネットワークから受信され記憶媒体に格納されたり、あるいはコンテンツを格納したメモリカード、ＣＤ、ＤＶＤ等の記憶媒体を再生専用器、あるいはゲーム機器に装着することにより、コンテンツ再生処理、あるいはプログラム実行が可能となる。

【０００３】

コンテンツの記憶素子として、最近多く利用される素子にフラッシュメモリがある。フラッシュメモリは、ＥＥＰＲＯＭ(Electrically Erasable Programmable ROM)と呼ばれる電気的に書き換え可能な不揮発性メモリの一形態である。従来のＥＥＰＲＯＭは、１ビットを２個のトランジスタで構成するために、１ビット当たりの占有面積が大きく、集積度を高くするのに限界があったが、フラッシュメモリは、全ビット一括消去方式により１ビットを１トランジスタで実現することが可能となった。フラッシュメモリは、磁気ディスク、光ディスク等の記録媒体に代わりうるものとして期待されている。

【０００４】

フラッシュメモリをデータ記録／再生機器に対して着脱自在に構成したメモリカードも知られている。このメモリカードを使用すれば、従来のＣＤ（コンパクトディスク：登録商標）、ＭＤ（ミニディスク：登録商標）等のディスク状媒体に換えてメモリカードを使用するディジタルオーディオ記録／再生装置を実現することができる。

【０００５】

このような、フラッシュメモリを使用したコンテンツ記憶素子をパーソナルコンピュータ（ＰＣ）、再生器等において使用する場合、ＦＡＴ(File Allocation Table)システムと呼ばれるファイル管理システムがアクセス情報テーブルとして一般的に使用される。ＦＡＴシステムでは、必要なファイルが定義されると、その中に必要なパラメータがファイルの先頭から順番にセットされる。その結果、ファイルサイズを可変長とすることができ、１ファイルを１または複数の管理単位（セクタ、クラスタ等）で構成することができる。この管理単位の関連事項がＦＡＴと呼ばれるテーブルに書かれる。このＦＡＴシステムは、記録媒体の物理的特性と無関係に、ファイル構造を容易に構築することができる。従って、ＦＡＴシステムは、フロッピーディスク、ハードディスクのみならず、光磁気ディスクにおいても採用することができる。上述したメモリカードにおいても、ＦＡＴシステムが採用されている。

【０００６】

音楽データ、画像データ、あるいはプログラム等の様々なコンテンツは、再生

機器として利用される再生装置、ゲーム機器、ＰＣ等の情報機器本体からのユーザ指示、あるいは接続された入力手段を介したユーザの指示により、上述のＦＡＴに基づいて例えば上述したフラッシュメモリから呼び出され、情報機器本体、あるいは接続されたディスプレイ、スピーカ等を通じて再生される。

【０００７】

さらに、ゲームプログラム、音楽データ、画像データ等、多くのソフトウエア・コンテンツは、一般的にその作成者、販売者に頒布権等が保有されている。従って、これらのコンテンツの配布に際しては、一定の利用制限、すなわち、正規のユーザに対してのみ、ソフトウエアの使用を許諾し、許可のない複製等が行われないようにする、すなわちセキュリティを考慮した構成をとるのが一般的となっている。

【０００８】

ユーザに対する利用制限を実現する１つの手法が、配布コンテンツの暗号化処理である。すなわち、例えばインターネット等を介して暗号化された音声データ、画像データ、ゲームプログラム等の各種コンテンツを配布するとともに、正規ユーザであると確認された者に対してのみ、配布された暗号化コンテンツを復号する手段、すなわち復号鍵を付与する構成である。

【０００９】

暗号化データは、所定の手続きによる復号処理によって利用可能な復号データ（平文）に戻すことができる。このような情報の暗号化処理に暗号化鍵を用い、復号処理に復号鍵を用いるデータ暗号化、復号化方法は従来からよく知られている。

【００１０】

【発明が解決しようとする課題】

例えばメモリ上にコンテンツを格納する場合、著作権などを保護するためにコンテンツ部に対する暗号化を行う場合があるが、コンテンツ部全体に対して一つの暗号化鍵を使って暗号化すると、同一の鍵のみに基づく大量の暗号文が発生し、攻撃が容易となってしまう危険性がある。コンテンツ部をできるだけ分割し、それぞれを異なる鍵で暗号化する方が望ましいと言える。コンテンツ暗号化の最

小単位として、セクタが挙げられる。

【００１１】

一方、複数のコンテンツ暗号鍵を安全に格納するための構成が問題となる。本発明は、コンテンツの暗号化等に適用する複数の鍵を安全に、すなわち、セキュリティを高めて保持することを可能としたデータ記憶装置、およびデータ記録方法、データ再生方法、並びにプログラム提供媒体を提供することを目的とする。

【００１２】

具体的には、本発明は、各コンテンツの属性情報であるヘッダの中に例えば、メディアの１ブロックあたりのセクタ数Ｍに対応するＭ個の鍵情報を暗号化して格納し、これらの暗号化処理をよりセキュリティの高い暗号化モードで実行する構成とすることにより、セキュリティを高い鍵保管を可能としたデータ記憶装置、およびデータ記録方法、データ再生方法、並びにプログラム提供媒体を提供する。

【００１３】

【課題を解決するための手段】

本発明の第１の側面は、

各々が予め定められたデータ容量を持つ複数のセクタを１ブロックとしたブロックを複数有するデータ格納領域と、暗号処理手段とを有するデータ記憶装置において、

前記暗号処理手段は、

前記データ記憶装置とデータ通信可能なデバイスから、前記データ格納領域に格納するデータの暗号処理鍵として、前記複数のセクタ各々の暗号処理用鍵または復号処理用鍵として対応づけられた複数の鍵からなる鍵セットを受領し、該受領した鍵セットに対してＣＢＣモードでの暗号処理を実行して、前記デバイスに送信する処理を実行する構成を有することを特徴とするデータ記憶装置にある。

【００１４】

さらに、本発明のデータ記憶装置の一実施態様において、前記暗号処理手段は、前記鍵セットに対するＣＢＣモードでの暗号処理を、前記データ記憶装置に固有の保存鍵Ｋｓｔｏを適用して実行して前記データ格納領域に格納するデータの

ヘッダ情報としての鍵データを生成する構成を有することを特徴とする。

【００１５】

さらに、本発明のデータ記憶装置の一実施態様において、前記データ記憶装置は、前記データ記憶装置とデータ通信可能な前記デバイスとの相互認証処理を実行する構成を有し、前記デバイスから受領する鍵セットは、前記相互認証時に生成したセッション鍵を用いたＣＢＣモードによる暗号化を行なったセッション鍵適用ＣＢＣモード処理鍵セットであり、前記暗号処理手段は、前記セッション鍵適用ＣＢＣモード処理鍵セットを前記セッション鍵を用いたＣＢＣモードで復号し、復号した鍵セットを前記データ記憶装置に固有の保存鍵Ｋｓｔｏに基づいてＣＢＣモードで暗号処理を実行して、保存鍵適用ＣＢＣモード処理鍵セットを生成して前記デバイスにヘッダ情報構成データとして送信する処理を実行する構成を有することを特徴とする。

【００１６】

さらに、本発明のデータ記憶装置の一実施態様において、前記データ記憶装置は、前記データ記憶装置とデータ通信可能な前記デバイスとの相互認証処理を実行する構成を有し、前記デバイスから受領する鍵セットは、前記データ格納領域に格納するデータのヘッダ情報として格納された鍵セットであって、前記データ記憶装置に固有の保存鍵Ｋｓｔｏに基づいてＣＢＣモードで暗号処理を実行した保存鍵適用ＣＢＣモード処理鍵セットであり、前記暗号処理手段は、前記保存鍵適用ＣＢＣモード処理鍵セットを保存鍵を用いたＣＢＣモードで復号し、復号した鍵セットを前記相互認証時に生成したセッション鍵に基づいてＣＢＣモードで暗号処理を実行して、セッション鍵適用ＣＢＣモード処理鍵セットを生成して前記デバイスにデータ復号用鍵情報構成データとして送信する処理を実行する構成を有することを特徴とする。

【００１７】

さらに、本発明のデータ記憶装置の一実施態様において、前記暗号処理手段は、前記データ記憶装置とデータ通信可能なデバイスから、前記データ格納領域に格納するデータの暗号処理鍵として、前記複数のセクタ各々の暗号処理用鍵または復号処理用鍵として対応づけられた複数の鍵からなる鍵セットと、前記複数の

セクタの少なくともいずれかに格納されるデータの改竄チェック値（ＩＣＶ）生成用の鍵Ｋｉｃｖと、を受領し、該受領した鍵セットに対してＣＢＣモードでの暗号処理を実行して、前記デバイスに送信する処理を実行する構成を有することを特徴とする。

【００１８】

さらに、本発明の第２の側面は、

各々が予め定められたデータ容量を持つ複数のセクタを１ブロックとしたブロックを複数有するデータ格納領域と、暗号処理手段とを有するデータ記憶装置と、

該データ記憶装置にデータを送信してデータの格納処理を実行するデータ記録装置とを有するデータ処理装置におけるデータ記録方法において、

前記データ記憶装置および前記データ記録装置間において相互認証処理を実行するステップと、

前記相互認証の成立を条件として、前記データ記録装置において、前記複数のセクタ各々の格納データの暗号処理に適用する複数の暗号鍵からなる暗号鍵セットを前記相互認証時に生成したセッション鍵を用いたＣＢＣモードによって暗号化して、セッション鍵適用ＣＢＣモード処理鍵セットを生成して、前記データ記憶装置に対して送信するステップと、

前記データ記憶装置において、前記セッション鍵適用ＣＢＣモード処理鍵セットをセッション鍵を適用したＣＢＣモードで復号するステップと、

復号した鍵セットを前記データ記憶装置に固有の保存鍵Ｋｓｔｏに基づいてＣＢＣモードで暗号化して、保存鍵適用ＣＢＣモード処理鍵セットを生成して、前記データ記録装置に送信するステップと、

前記データ記録装置において、前記データ記憶装置に格納するデータに対応するヘッダ情報として、受信した前記保存鍵適用ＣＢＣモード処理鍵セットデータを構成要素として含むヘッダ情報を生成するステップと、

を有することを特徴とするデータ記録方法にある。

【００１９】

さらに、本発明の第３の側面は、

各々が予め定められたデータ容量を持つ複数のセクタを１ブロックとしたブロックを複数有するデータ格納領域と、暗号処理手段とを有するデータ記憶装置と、

該データ記憶装置からデータを受信してデータの再生処理を実行するデータ再生装置とを有するデータ処理装置におけるデータ再生方法において、

前記データ記憶装置および前記データ再生装置間において相互認証処理を実行するステップと、

前記相互認証の成立を条件として、前記データ格納領域に格納されたデータのヘッダ情報に含まれる鍵セットであって、前記データ記憶装置に固有の保存鍵Ｋｓｔｏに基づいてＣＢＣモードで暗号処理を実行した保存鍵適用ＣＢＣモード処理鍵セットを前記データ記録装置から前記データ記憶装置に送信するステップと、

前記データ記憶装置において、

前記保存鍵適用ＣＢＣモード処理鍵セットを保存鍵を用いたＣＢＣモードで復号するステップと、

復号した鍵セットを前記相互認証時に生成したセッション鍵に基づいてＣＢＣモードで暗号処理を実行して、セッション鍵適用ＣＢＣモード処理鍵セットを生成して前記データ再生装置に送信するステップと、

前記データ再生装置において、前記セッション鍵適用ＣＢＣモード処理鍵セットをセッション鍵を適用したＣＢＣモードにより復号して、前記データ格納領域の各セクタに格納された暗号化セクタデータの復号用の鍵セットを取得するステップと、

を有することを特徴とするデータ再生方法にある。

【００２０】

さらに、本発明の第４の側面は、

各々が予め定められたデータ容量を持つ複数のセクタを１ブロックとしたブロックを複数有するデータ格納領域と、暗号処理手段とを有するデータ記憶装置と、

該データ記憶装置にデータを送信してデータの格納処理を実行するデータ記録

装置とを有するデータ処理装置におけるデータ記録処理をコンピュータ・システム上で実行せしめるコンピュータ・プログラムを提供するプログラム提供媒体であって、前記コンピュータ・プログラムは、

前記データ記憶装置および前記データ記録装置間において相互認証処理を実行するステップと、

前記相互認証の成立を条件として、前記データ記録装置において、前記複数のセクタ各々の格納データの暗号処理に適用する複数の暗号鍵からなる暗号鍵セットを前記相互認証時に生成したセッション鍵を用いたＣＢＣモードによって暗号化して、セッション鍵適用ＣＢＣモード処理鍵セットを生成して、前記データ記憶装置に対して送信するステップと、

前記データ記憶装置において、前記セッション鍵適用ＣＢＣモード処理鍵セットをセッション鍵を適用したＣＢＣモードで復号するステップと、

復号した鍵セットを前記データ記憶装置に固有の保存鍵Ｋｓｔｏに基づいてＣＢＣモードで暗号化して、保存鍵適用ＣＢＣモード処理鍵セットを生成して、前記データ記録装置に送信するステップと、

前記データ記録装置において、前記データ記憶装置に格納するデータに対応するヘッダ情報として、受信した前記保存鍵適用ＣＢＣモード処理鍵セットデータを構成要素として含むヘッダ情報を生成するステップと、

を有することを特徴とするプログラム提供媒体にある。

【００２１】

さらに、本発明の第５の側面は、

各々が予め定められたデータ容量を持つ複数のセクタを１ブロックとしたブロックを複数有するデータ格納領域と、暗号処理手段とを有するデータ記憶装置と、

該データ記憶装置からデータを受信してデータの再生処理を実行するデータ再生装置とを有するデータ処理装置におけるデータ再生処理をコンピュータ・システム上で実行せしめるコンピュータ・プログラムを提供するプログラム提供媒体であって、前記コンピュータ・プログラムは、

前記データ記憶装置および前記データ再生装置間において相互認証処理を実行

するステップと、

前記相互認証の成立を条件として、前記データ格納領域に格納されたデータのヘッダ情報に含まれる鍵セットであって、前記データ記憶装置に固有の保存鍵Ｋｓｔｏに基づいてＣＢＣモードで暗号処理を実行した保存鍵適用ＣＢＣモード処理鍵セットを前記データ記録装置から前記データ記憶装置に送信するステップと、

前記データ記憶装置において、

前記保存鍵適用ＣＢＣモード処理鍵セットを保存鍵を用いたＣＢＣモードで復号するステップと、

復号した鍵セットを前記相互認証時に生成したセッション鍵に基づいてＣＢＣモードで暗号処理を実行して、セッション鍵適用ＣＢＣモード処理鍵セットを生成して前記データ再生装置に送信するステップと、

前記データ再生装置において、前記セッション鍵適用ＣＢＣモード処理鍵セットをセッション鍵を適用したＣＢＣモードにより復号して、前記データ格納領域の各セクタに格納された暗号化セクタデータの復号用の鍵セットを取得するステップと、

を有することを特徴とするプログラム提供媒体にある。

【００２２】

なお、本発明の第４、５の側面に係るプログラム提供媒体は、例えば、様々なプログラム・コードを実行可能な汎用コンピュータ・システムに対して、コンピュータ・プログラムをコンピュータ可読な形式で提供する媒体である。媒体は、ＣＤやＦＤ、ＭＯなどの記録媒体、あるいは、ネットワークなどの伝送媒体など、その形態は特に限定されない。

【００２３】

このようなプログラム提供媒体は、コンピュータ・システム上で所定のコンピュータ・プログラムの機能を実現するための、コンピュータ・プログラムと提供媒体との構造上又は機能上の協働的関係を定義したものである。換言すれば、該提供媒体を介してコンピュータ・プログラムをコンピュータ・システムにインストールすることによって、コンピュータ・システム上では協働的作用が発揮され

、本発明の他の側面と同様の作用効果を得ることができるのである。

【００２４】

本発明のさらに他の目的、特徴や利点は、後述する本発明の実施例や添付する図面に基づくより詳細な説明によって明らかになるであろう。

【００２５】

【発明の実施の形態】

［システム概要］

図１に本発明のデータ処理装置の適用可能なコンテンツ配信システム構成を示す。例えば音楽データ、画像データ、その他各種プログラム等のコンテンツが、コンテンツ保持者またはサービスプロバイダのようなシステム運営者１０１から、インターネット等のネットワークを介して、またはＣＤ、ＤＶＤ、フラッシュメモリを搭載したメモリカード等の各種記録媒体であるメディア１０３に格納され、デバイス１０２に受信または装着されて再生、実行される。デバイスは、例えばパーソナルコンピュータ（ＰＣ）、再生専用器、ゲーム器等のコンテンツ再生機能を有するデバイスであり、例えば画像コンテンツを表示する表示装置、ユーザの指示を入力する入力装置を有する。

【００２６】

このようなコンテンツ配信システムの構成中、コンテンツを再生するデバイスと、コンテンツを格納するメディアとの詳細構成を図２に示す。

【００２７】

図２は、デバイス２００、メディア１，２１０、メディア２，２３０の詳細構成を示している。メディア１，２１０は、単純なデータ読み出し、書き込み処理のみをサポートする制御部を持つメディアであり、メディア２，２３０は、メディアを装着するデバイスとの相互認証処理を実行し、またメディアに格納するコンテンツの暗号処理を実行するコントローラを有するメディアである。メディア１，２１０、メディア２，２３０の双方ともデバイス２００に対する装着が可能である。

【００２８】

図２のデバイス２００は、インターネット等のデータ通信手段を介したデータ

送受信処理を実行する通信部２０１、各種指示を入力する入力部２０２、メッセージ、コンテンツ等の表示を実行する表示部２０３、これらの制御を実行する制御部２０５と、メディアとのデータ入出力処理のインタフェース機能を持つメモリインタフェース（Ｉ／Ｆ）部３００とを持つデバイスコントローラ２０４、さらに、コンテンツのファイル群と、不正なメディアやコンテンツの失効情報としてのリボケーションリストを格納している内部メモリとしてのメモリ部２０７を有する。なお、内部メモリ内に格納されるリボケーションリスト等のデータファイルは、ファイル割り当てテーブルによって管理され読み出し可能な構成を持つ。

【００２９】

デバイス２００は、コンテンツの再生時に再生対象のコンテンツがリボケーションリストに格納された失効メディア、失効コンテンツに対応していないことを確認した上で再生を行なう。再生対象のコンテンツがリボケーションリストにリストアップされていた場合は、再生エラーとなり、再生処理が実行されない。リボケーションリスト、およびリボケーションリストを適用した再生処理については後段で詳細に説明する。

【００３０】

メディア１，２１０は、データ入出力を制御する制御部２１１と、コンテンツを格納するメモリ部２１２を有し、メモリ部２１２は、コンテンツを対応ヘッダ情報とともに格納するのみならず、メディア個々に固有の識別情報としてのメディアＩＤ、さらに、メモリアクセスコントロール情報を記述したアクセス許可テーブルであるＢＰＴ（Block Permission Table）を格納している。

【００３１】

デバイス２００のファイルシステムはメディアを認識した後に、アクセス許可テーブルであるＢＰＴをメディアから読み込み、メディアへ直接アクセスを行うメモリインターフェイス部３００にＢＰＴを転送し、管理させる。メモリインターフェイス部３００は、ＢＰＴを受信した後、受信したＢＰＴについて改竄チェック値（ＩＣＶ）の検証を行う。ＩＣＶが正当なものと判断された場合のみ、ＢＰＴを有効なものとして保存する。メモリインターフェイス部３００は、メディ

アのメモリにアクセスする命令を受信した時、このメディアのＢＰＴに基づいたアクセスのみ実行する。ＢＰＴの構成、およびＢＰＴを用いた処理に関しては後段で詳細に説明する。

【００３２】

メディア２，２３０は、コントローラ２３１と、メモリ部２３２によって構成され、メモリ部２３２は、コンテンツを対応ヘッダ情報とともに格納し、さらにアクセス許可テーブルであるＢＰＴ（Block Permission Table）を格納している。コントローラ２３１は、メモリ部２３２に対するデータ格納、またはデータ読み出し用インタフェースとしてのメモリインタフェース（Ｉ／Ｆ）部２３４、メデイアの識別子としてのメディア２ＩＤ、相互認証処理に適用する認証鍵Ｋａｋｅ、コンテンツのメモリ部２３２への保存時の暗号鍵である保存鍵Ｋｓｔｏ、さらに暗号化対象の鍵を暗号化する時の初期値ＩＶ＿ｋｅｙｓ等を格納した内部メモリ２３５、認証処理あるいはコンテンツの暗号化、復号処理を実行し、レジスタを備えた暗号処理部２３６、そして、これら各部の制御を実行する制御部２３３とを有する。

【００３３】

［メディア内メモリ構成］

次に、メディア１，２１０、メディア２，２３０の各メモリ部のデータ格納構成を図３に示す。メモリ部は例えば、ＥＥＰＲＯＭ(Electrically Erasable Programmable ROM)と呼ばれる電気的に書き換え可能な不揮発性メモリの一形態であるフラッシュメモリであり、ブロック単位の一括消去方式によるデータ消去が実行される。

【００３４】

図３（ａ）に示すように、フラッシュメモリは、第1～Ｎまでの複数ブロックを有し、各ブロックは、（ｂ）に示すように第1～Ｍまでの複数セクタによって構成され、各セクタは（ｃ）に示すように実データを含むデータ部と、エラー訂正コード等の冗長データを含む冗長部によって構成される。後段で詳細に説明するが、冗長部には各セクタのデータ部内のセクタデータ改竄チェック値としてのＩＣＶが格納される場合がある。

【００３５】

[主要コマンド]

次に図２のデバイス２００において、制御部２０５と、メモリインタフェース（Ｉ／Ｆ）部３００において発行される主なコマンドについて説明する。

【００３６】

まず、制御部２０５からメモリインターフェイス（Ｉ／Ｆ）部３００に対するコマンドには、以下のものがある。

・ステータス読み出しコマンド

現在のメモリインタフエース内のステータスを設定したステータスレジスタの状態の読み出し。メモリインターフェイス（Ｉ／Ｆ）部３００は、ステータスレジスタの内容を返す。

・セクタ読み出しコマンド

指定したセクタのデータ読み出し処理命令。

・セクタ書き込みコマンド

指定したセクタへのデータ書き込み処理命令。

・セクタ復号読み出しコマンド

セットされたヘッダの情報を元に、指定されたセクタの暗号化データを復号して読み出す処理の実行命令。

・セクタ暗号書き込みコマンド

セットされたヘッダの情報を元に、指定されたセクタへデータを暗号化して書き込む処理の実行命令。

・ヘッダ生成コマンド

指定されたパラメータを元にヘッダを生成する処理の実行命令。

・ヘッダセットコマンド

ヘッダをメモリーインターフェイス内にセットする処理の実行命令。

・ＢＰＴセットコマンド

ＢＰＴをメモリーインターフェイス内にセットする処理の実行命令。

・リボケーションリスト（Revocation List）セットコマンド

不正メディア、不正コンテンツのリストであるリボケーションリスト（Revoca

tion List）をメモリーインターフェイス内にセットする処理の実行命令。

・更新用リボケーションリスト（Revocation List）チェックコマンド

更新用リボケーションリスト（Revocation List）に現在のリボケーションリスト（Revocation List）を更新してよいかチェックする処理の実行命令。

・メディア１認識コマンド

接続されたメディア１に対してメディアの識別子（ＩＤ）を読み出して、そのＩＤが有効かどうかチェックする処理の実行命令。

・メディア２認識コマンド

接続されたメディア２に対して相互認証をして、メディアの識別子（ＩＤ）が有効かどうかチェックする処理の実行命令。

・ファイル割り当てテーブル呼び出しコマンド

メモリ内のファイル割り当てテーブルを読み出す処理の実行命令。

・ファイル割り当てテーブル更新コマンド

メモリへのファイル割り当てテーブルを更新する処理の実行命令。

【００３７】

メモリインターフェイス（Ｉ／Ｆ）部３００からメディア１に対するコマンドは、以下のものがある。

・ＩＤ読み出しコマンド

メディア１の持つＩＤを読み出す処理の実行命令。

【００３８】

[デバイス内メモリインタフェース詳細構成]

次にデバイス２００のメモリインタフェース（Ｉ／Ｆ）部３００の詳細構成を図４に示す。各構成部の機能を説明する。

【００３９】

・ステータスレジスタ３０１

メモリインターフェイスの内部ステータスを保存するレジスタである。ステータスレジスタ３０１の構成例を図５に示す。各ビットは以下の意味を持つ。

・ビット０（bit 0）：ビジーフラグ　（１：ヒジー（busy），０：待機（ready））

メモリインターフェイスが内部処理をしているかの判別用ビットである。

・ビット１（bit 1）:読み出し成功フラグ　(1:成功（success），0:失敗（fail））

メモリからデータの読み出しが成功したかの判別用ビットである。

・ビット２（bit 2）:書き込み成功フラグ　(1: 成功（success），0:失敗（fail））

メモリへデータの書き込みが成功したかの判別用ビットである。

・ビット３（bit 3）:メディア１セットフラグ　(1: セット（set），0: 未セット（not set））

接続されたメディア１が利用可能かの判別用ビットである。

・ビット４（bit 4）:メディア２セットフラグ　(1: セット（set），0: 未セット（not set））

接続されたメディア２が利用可能かの判別用ビットである。

・ビット５（bit 5）:メディア１有効フラグ　(1: 有効（OK），0: 無効（NG））

接続されたメディア１の識別子（ＩＤ）が、リボケーションリスト（Revocation List）内のリボーク（排除）メディア対象外かの判別用ビットである。

・ビット６（bit 6）:メディア２有効フラグ　(1: 有効（OK），0: 無効（NG））

接続されたメディア２の識別子（ＩＤ）が、リボケーションリスト（Revocation List）内のリボーク（排除）メディア対象外かの判別用ビットである。

・ビット７（bit 7）: ヘッダセット成功フラグ　(1: 成功（success），0:失敗（fail））

ヘッダがメモリインターフェイス内にセット出来たかの判別用ビットである。

・ビット８（bit 8）:ヘッダ生成成功フラグ　(1: 成功（success），0:失敗（fail））

ヘッダの生成が成功したかの判別用ビットである。

・ビット９（bit 9）:リボケーションリスト（Revocation List）セットフラグ　(1: セット（set），0: 未セット（not set））

リボケーションリスト（Revocation List）がメモリインターフェイス内にセット出来たかの判別用ビットである。

・ビット１０（bit 10）:更新用リボケーションリスト（Revocation List）有効フラグ（1: 有効（OK），0: 無効（NG））

更新用リボケーションリスト（Revocation List）が有効であるかどうかの判別用ビットである。

【００４０】

ステータスレジスタ３０１は、これらのインタフェース（Ｉ／Ｆ）部３００のステータス情報を保持する。

【００４１】

図４に戻り、各構成の機能について説明を続ける。

・コマンドレジスタ３０２

制御部より送信されたコマンドを保存するレジスタ

・アドレスレジスタ３０３

データの転送開始セクタを設定するレジスタ

・カウントレジスタ３０４

データの全転送セクタ数を設定するレジスタ

【００４２】

なお、外部メモリ、内部メモリに対するデータの読み書きは、アドレスレジスタに読み書きを開始するセクタアドレスを設定し、カウントレジスタに読み書きをする総セクタ数を設定し、コマンドレジスタにセクタ読み書きコマンドをセットすることで実行される。

【００４３】

・コントロールレジスタ３０５

メモリインターフェイスの動作を設定するレジスタ

・送受信制御部３０６

各種レジスタおよび送受信バッファなど、メモリインターフェイスの制御を行う。

・送信バッファメモリ３０７

送信データを格納するバッファ

・受信バッファメモリ３０８

受信データを格納するバッファ

・送信レジスタ３０９

送信バッファメモリ３０７内のデータを送信するためのレジスタ

・受信レジスタ３１０

受信したデータを保存し受信バッファメモリ３０８に転送するためのレジスタ

【００４４】

・暗号処理部３２０

送信バッファメモリ３０７、受信バッファメモリ３０８内のデータに対して、各種暗号処理を施す。

・メモリ部３２１

暗号処理部３２０における暗号処理に必要な鍵情報、および内部メモリから読み込まれるリボケーションリスト、外部メモリから読み込まれるアクセス許可テーブルとしてのブロック・パーミッション・テーブル（ＢＰＴ）を格納、保存する領域である。リボケーションリスト、ブロック・パーミッション・テーブル（ＢＰＴ）それぞれがメモリインタフェース内に有効にセットされた場合、送受信制御部３０６が制御部からのメディア認識コマンド、あるいは外部メモリに対するデータの読み書きコマンド等を受信した場合、セットされたリボケーションリスト、ブロック・パーミッション・テーブル（ＢＰＴ）を参照した処理が実行される。これらの処理については、後段でフローを用いて詳細に説明する。

【００４５】

さらに、メモリ部３２１には、暗号処理に必要な鍵情報としては、以下のデータが格納される。

Ｋdist：メディア２に格納されるコンテンツ以外のコンテンツのセキュリテイヘッダ（Security Header）に含まれる配送鍵。コンテンツＩＣＶ生成鍵Ｋicv_cont、コンテンツ鍵Ｋcを暗号化する。

Ｋicv_sh ：セキュリテイヘッダ（Security Header）のＩＣＶを生成する際に用いるセキュリティヘッダＩＣＶ生成鍵。

ＩＶsh：セキュリテイヘッダ（Security Header）のＩＣＶを生成する際に用いる初期値（ＩＶ：Initial Value）。

ＭＫake：相互認証用のマスター鍵。

ＩＶake：相互認証用の鍵の生成処理に適用するための初期値（ＩＶ：Initial Value）。

ＩＶauth：相互認証時のデータ生成用の初期値(ＩＶ：Initial Value)。

ＭＫicv_rl：リボケーションリスト（Revocation List）のＩＣＶ鍵を生成するマスター鍵。

ＩＶicv_rl ：リボケーションリスト（Revocation List）のＩＣＶ鍵を生成する時の初期値（ＩＶ：Initial Value）。

ＩＶrl：リボケーションリスト（Revocation List）のＩＣＶ生成時に用いる初期値（ＩＶ：Initial Value）。

ＩＶ_keys：メディア2で、コンテンツ暗号化用の鍵を暗号化する時の初期値（ＩＶ：Initial Value）。

ＭＫicv_bpt:アクセス許可情報であるＢＰＴ（Block Permission Table）のＩＣＶ鍵を生成するマスター鍵。

ＩＶicv_bpt: アクセス許可情報であるＢＰＴ（Block Permission Table）のＩＣＶ鍵を生成する時のＩＣＶ生成時に用いる初期値（ＩＶ：Initial Value）。

ＩＶbpt：アクセス許可情報であるＢＰＴ（Block Permission Table）の初期値（ＩＶ：Initial Value）。

【００４６】

・ＥＣＣ回路３２３

送信レジスタ３０９、受信レジスタ３１０にあるデータについて、ＥＣＣチェックを行う専用ブロックである。

【００４７】

・外部メモリ入出力インターフェイス３２４

外部メモリ（メディア１，２）に対する入出力インターフェイス。外部メモリとしては例えばフラッシュメモリを搭載したメモリカード等がある。例えばコン

テンツ、およびコンテンツ記録再生に伴うヘッダ情報、さらにブロック・パーミッション・テーブル（ＢＰＴ）がこの外部メモリ入出力インターフェイスを介して入出力する。

・内部メモリ入出力インターフェイス３２５

内部メモリに対する入出力インターフェイス。当インタフェースを介して、内部メモリに格納された例えばリボケーションリストの入出力が実行される。

【００４８】

外部メモリ入出力インターフェイス３２４、および内部メモリ入出力インターフェイス３２５からは、処理に応じて以下の各信号が外部メモリ（メディア１，２）、あるいは内部メモリに対して出力される。

ＣＬＥ:コマンドラッチイネーブル

ＡＬＥ:アドレスラッチイネーブル

ＣＥ:チップイネーブル

ＷＥ:ライトイネーブル

ＲＥ:リードイネーブル

また、外部メモリ（メディア１，２）、あるいは内部メモリからの信号として

ＷＰ:ライトプロテクト　（外部メモリ（メディア１、２）にのみ適用）

ＲＤＹ／ＢＵＳＹ:レディー・ビジー

これら各種信号が入力される。

【００４９】

［メモリ格納コンテンツ構成］

次に、メディアのフラッシュメモリに格納されるコンテンツ構成について図６を用いて説明する。音楽データ、画像データ等、各コンテンツは、図６（ａ）に示すように各種属性情報からなるセキュリティヘッダと、実データ部としてのコンテンツとによって構成される。

【００５０】

メディアのフラッシュメモリは、図６（ｂ）に示すように、複数コンテンツのセキュリティヘッダ部とコンテンツ部との各ペアを格納する。前述したように、

フラッシュメモリはブロック単位で消去が実行されるので、１ブロックには同一コンテンツに関するセキュリティヘッダ部またはコンテンツ部を格納する形態とし、一括した消去処理が許容される場合を除いて、異なるコンテンツを１つのブロックに格納する処理は行なわない。

【００５１】

［セキュリティヘッダ構成］

セキュリティヘッダは、各コンテンツに対応する属性情報である。セキュリティヘッダのデータ構成を図７に示す。各データ内容について説明する。

【００５２】

・フォーマットバージョン（Format Version）

セキュリティヘッダ（Security Header）のフォーマットバージョンを示す。

・コンテンツＩＤ（Content ID）

コンテンツの識別子（ＩＤ）を示す。

・コンテンツタイプ（Content Type）

コンテンツの種類を示す。例えばメディア１、またはメディア２に格納されたコンテンツ、あるいは放送コンテンツ等である。

・データタイプ（Data Type）

コンテンツの属性、例えば音楽、画像等のデータであるか、プログラムであるか等を示す。

・暗号アルゴリズム（Encryption Algorithm）

コンテンツのコンテンツ鍵(Kc)を使った暗号化処理アルゴリズムを示す。例えばＤＥＳによる暗号化であるかトリプルＤＥＳ（Triple-DES)によるか等を示す。

・暗号化モード（Encryption Mode）

暗号化アルゴリズム（Encryption Algorithm）で指定されたアルゴリズムに対応する暗号モードを示す。例えばＥＣＢモードかＣＢＣモードか等を示す。

【００５３】

・暗号化フォーマットタイプ（Encryption Format Type）

コンテンツの暗号化フォーマットを示す。タイプ１かタイプ２か、

コンテンツ全体に対して１つのコンテンツ鍵Ｋｃで暗号化するタイプをタイプ１とし、コンテンツのセクタ毎に異なる鍵Ｋｓｅｃ＿ｎを適用してコンテンツの暗号化を行なう態様をタイプ２とする。

【００５４】

図８に各タイプの暗号化フォーマット構成を示す。図８（ａ）がタイプ１の暗号化フォーマットで暗号化されたコンテンツのメモリ格納構成であり、（ｂ）がタイプ２の暗号化フォーマットで暗号化されたコンテンツのメモリ格納構成である。

【００５５】

図８（ａ）に示すタイプ1の暗号化フォーマットは、コンテンツがすべて１つのコンテンツ鍵Ｋｃを用いて暗号化されてメモリに格納された構成、すなわちセクタ非依存型暗号化処理である。図８（ｂ）に示すタイプ２の暗号化フォーマットは、フラッシュメモリの各セクタ毎に異なるセクタ鍵Ｋｓｅｃ＿１～Ｋｓｅｃ＿ｍが適用されて暗号化されたコンテンツが格納された構成、すなわちセクタ依存型暗号化処理である。例えば図８（ｂ）のフラッシュメモリのセクタ１では、セクタ１の暗号化鍵としてＫｓｅｃ＿１が対応して設定され、セクタ１に格納されるコンテンツは、各ブロックにおいて、すべてＫｓｅｃ＿１を適用した暗号化処理が施されて格納される。フラッシュメモリのセクタｍでは、セクタｍの暗号化鍵としてＫｓｅｃ＿ｍが対応して設定され、セクタｍに格納されるコンテンツは、各ブロックにおいて、すべてＫｓｅｃ＿ｍを適用した暗号化処理が施されて格納される。

【００５６】

このように、本発明の構成においては、各セクタ毎に異なる暗号化鍵を適用したコンテンツの暗号処理が適用される。さらに、各セクタ毎に異なる暗号化鍵を適用した処理態様においても、１つのセクタに１つの鍵を適用したシングルＤＥＳによる処理、１つのセクタに複数の鍵を適用したトリプルＤＥＳによる処理等、各種の暗号化態様が適用可能である。これらの処理形態については、さらに後段で詳細に説明する。

【００５７】

図７に戻り、セキュリティヘッダの構成について説明を続ける。

・暗号化フラグ（Encryption Flag）

ブロック内の各セクタの暗号化・非暗号化を示すフラグ。ブロック内のセクタ数（例えば３２セクタ）分のフラグを持つ。例えば０：非暗号化セクタ、１：暗号化セクタ。なお、本例では１ブロックを３２セクタとする。

【００５８】

・ＩＣＶフラグ（ICV Flag）

ブロック内の各セクタのＩＣＶ付加・非付加を示すフラグ。ブロック内のセクタ数（32セクタ）分のフラグを持つ。例えば０：ＩＣＶなし、１：ＩＣＶあり

【００５９】

・暗号化コンテンツ鍵（Ｋｃ_Encrypted 0-31）

暗号化されたコンテンツ鍵の格納領域(32個)

・暗号化ＩＣＶ生成鍵（Ｋicv_cont_encrypted)

暗号化されたコンテンツのＩＣＶ作成のための鍵の格納領域

【００６０】

・有効リボケーションリストバージョン（Valid Revocation List version）

コンテンツ再生の際に有効に適用されるリボケーションリスト（Revocation List）のバージョン。

コンテンツ再生の際に、セットされているリボケーションリスト（Revocation List）のバージョンがこれより古い場合、再生を許可しない。なお、自己デバイスにおいて格納したデータの再生処理等、リボケーションリストの参照を適用する必要がないコンテンツには０を設定する。

【００６１】

・セキュリティヘッダＩＣＶ（ICV of Security Header）

セキュリティヘッダ（Security Header）の改竄チェック値（ＩＣＶ）。

【００６２】

[リボケーションリスト]

次に、不正なメディアやコンテンツの失効情報としてのリボケーションリストの構成について説明する。図９にリボケーションリストの構成を示す。以下、各

データについて説明する。

【００６３】

・リボケーションリスト識別子（Revocation List ID）

リボケーションリスト（Revocation List）固有の識別子としてのＩＤである。

【００６４】

・リボケーションリストバージョン（Revocation List Version）

リボケーションリスト(Revocation List)のバージョンを示す。リボケーションリストは、更新され、更新時に新たな不正なメディアやコンテンツの失効情報を追加する。

【００６５】

本発明の構成では、リボケーションリスト(Revocation List)にバージョン情報を設定し、コンテンツのヘッダ内に有効なリボケーションリストのバージョン情報を設定する。コンテンツ読み出しの際に、現在デバイスに保持しているリボケーションリストのバージョンとコンテンツのヘッダ内にある有効なリボケーションリストのバージョンとを比較する。この際、現在保持しているリボケーションリストのバージョンの方がより古い場合には、コンテンツの読み出しを中止する。その結果、リボケーションリストの更新を行わなければ、そのコンテンツの読み出しはできない。

【００６６】

また、リボケーションリストの更新時にメモリ・インターフェース部が現在のリボケーションリストのバージョン情報と更新用のリボケーションリストのバージョン情報とを比較して、新しいリボケーションリストであると判断した時のみ、リボケーションリスト更新を許可する構成とする。

【００６７】

バージョン情報を用いたリボケーションリストの新旧比較処理、更新処理の具体的処理例については、処理フローを用いて後段で詳細に説明する。

【００６８】

・メディア１ＩＤ数（Number of Medial ID）

失効しているメディア１（Media1 ID）の総数

・メディア１ＩＤ（０）－メディア１ＩＤ（Ｌ－１）（Media1ID(0) - Media1 ID(L-1)）

失効しているメディア１の識別子のリストである。

【００６９】

・メディア２ＩＤ数（Number of Media2 ID）

失効しているメディア２（Media２ ID）の総数

・メディア２ＩＤ（０）－メディア２ＩＤ（Ｍ－１）（Media2ID(0) - Media2 ID(M-1)）

失効しているメディア２の識別子のリストである。

【００７０】

・コンテンツＩＤ数（Number of Contents ID）

失効しているコンテンツＩＤ（Contents ID）の総数

・コンテンツＩＤ（０）－コンテンツＩＤ（Ｎ－１）（Contents ID(0) - Contents ID(N-1)）

失効しているコンテンツ識別子のリストである。

【００７１】

・リボケーションリストＩＣＶ（ICV of Revocation List）

リボケーションリストの改竄チェック用のＩＣＶ

【００７２】

上述のように、本発明におけるリボケーションリストは、複数の種類（メディア、コンテンツ）の識別子（ＩＤ）から構成される。このように、コンテンツやメディアの失効情報であるリボケーションリスト（Revocation List）に複数の種類のリボーク対象ＩＤ、すなわちメディアＩＤ、コンテンツＩＤを設け，それぞれの照合を異なる動作として行うことによって，一つのリボケーションリストで複数のコンテンツ、メディアを排除することが可能となる。メディアの挿入時やコンテンツの読み出し時にメモリ・インターフェース部において、利用メディアまたは利用コンテンツの識別子（ＩＤ）と、リボケーションリストにリストされたＩＤとの照合を実行することにより、不正なメディアの使用や不正なコンテ

ンツの読み出しを禁止することができる。

【００７３】

このようにコンテンツやメディアの複数のＩＤを１つのリボケーションリストに設定した構成により１つのリボケーションリストで複数の種類のメディアとコンテンツのリボーク（排除）が可能になる。メディア起動時のリボケーションリストに基づくメディアの検証処理、コンテンツ処理時のコンテンツ検証処理の具体的処理については、後段で説明する。

【００７４】

また、本発明の構成では、リボケーションリストは、外部メモリ等に直接アクセスするメモリインタフェースにセットアップされ、セットアップ後は、メデイアの装着時、コンテンツの再生時においてメモリインタフェースにおいて継続的に利用可能な構成としたので、コンテンツの利用時に繰り返し内部メモリから読み出すなどの処理が不要となり処理が効率的に実行される。

【００７５】

[ブロック・パーミッション・テーブル（ＢＰＴ）]

次に、アクセス許可テーブルとして使用されるブロック・パーミッション・テーブル（ＢＰＴ：Block Permission Table）の構成について説明する。従来、例えばＰＣ等においてコンテンツの再生を実行する場合、ＰＣ内のＯＳのファイルシステムが主体的に、記録メディアに格納されているアクセス情報テーブル（例えば、File Allocation Table ; FAT）を読み込んで管理しており、ファイルシステムがそのアクセス情報テーブルの内容を自由に書き換えが出来た。その為に、書込み禁止を設定したアクセス情報テーブルを格納する記録メディアがあっても、そのアクセス情報テーブルをファイルシステムが読みとって書き換えることによって、記録メディア内のデータを書き換えられる可能性がある。

【００７６】

本発明のデータ処理装置において採用されるブロック・パーミッション・テーブル（ＢＰＴ）は、デバイスにおける書き替えを禁止したブロックに格納されるメディア自身のアクセス許可テーブルである。デバイスはＢＰＴを格納したメディアを用いて、コンテンツデータ書き込み等のデータ処理を実行する場合、メデ

ィアに直接アクセスするデバイスのメモリインターフェイス部にブロック・パーミッション・テーブル（ＢＰＴ）をセットすることで、デバイスの制御部がいかなるプログラムを実行中でも、メディアのアクセス許可テーブルであるブロック・パーミッション・テーブル（ＢＰＴ）に設定された許可情報に従ったメモリアクセスが行われる構成とした。

【００７７】

図１０にブロック・パーミッション・テーブル（ＢＰＴ）の構成を示す。以下、各データについて説明する。

【００７８】

・フォーマットバージョン（Format Version）

ＢＰＴ（Block Permission Table）のフォーマットバージョンを示す。ＢＰＴ自体にも、各種のフォーマットがあり、そのいずれであるかを識別するデータである。

・ＢＰＴ識別子（BPT ID）

ブロック・パーミッション・テーブル（ＢＰＴ：Block Permission Table）の識別子（ＩＤ）である。

・ブロック数（Number of Blocks）

ＢＰＴ（Block Permission Table）で扱うブロックの総数を示す。前述したように、フラッシュメモリはブロック毎の消去がなされる。ＢＰＴにより管理されるブロック数を示している。

・ブロック＃１－ブロック＃ｎ許可フラグ（Block #1 - #n Permission Flag）

各ブロックのアクセス制限フラグを示している。例えばフラグ０のブロックは、消去不可ブロックであり、フラグ１のブロックは消去可ブロックであることを示す。

・ＢＰＴ－ＩＣＶ（ICV of BPT）

ＢＰＴ（Block Permission Table）の改竄チェック用のＩＣＶである。

【００７９】

デバイスのファイルシステムはデバイスを認識した後に、ブロック・パーミッ

ション・テーブル（ＢＰＴ）を例えばフラッシュメモリを搭載したメモリカード等のメディアから読み込み、メディアへ直接アクセスを行うメモリインターフェイス部にＢＰＴを転送し、そのメディアに対するアクセス許可テーブルとして管理させる。メモリインターフェイス部は、アクセス許可テーブルを受信しＢＰＴをセット（ｅｘ．図４に示すメモリ部３２１）する。メモリインターフェイスは、メディアのメモリにアクセスする命令を受信した時、このメディアのアクセス許可テーブルに基づいたアクセスのみを実行する。

【００８０】

ブロック・パーミッション・テーブル（ＢＰＴ）には、例えばメディアのフラッシュメモリの各ブロック単位での許可された処理態様、具体的には例えば消去可ブロック、消去不可ブロック、あるいは再生可ブロック、再生不可ブロック等の設定がなされている。メモリインタフェースは、これらのＢＰＴ設定に従って処理の可否を決定する。これらの処理の詳細は、後段でさらに詳細に説明する。

【００８１】

なお、ブロック・パーミッション・テーブル（ＢＰＴ）には、改竄防止のための改竄チェック値ＩＣＶが設定され、ＢＰＴのメモリインタフェースへのセット時には、ＩＣＶチェックが実行され、改竄ありと判定された場合には、ＢＰＴのセット処理を実行しない。従って、不正なアクセス許可テーブルを作成して、使用することが防止される。ＢＰＴのＩＣＶはメディアの識別子（ＩＤ）に基づいて生成する。そのために、他のメディアにアクセス許可テーブルをコピーしたとしてもそのメディアは使用できない。ＩＣＶの生成については、後述する。

【００８２】

メディアは、その製造時にブロック・パーミッション・テーブル（ＢＰＴ）をメモリ（ｅｘ．フラッシュメモリ）の所定ブロックに書き込んで出荷する。この際、ブロック・パーミッション・テーブル（ＢＰＴ）を格納したメモリ内のブロックについては、ブロック消去不可の設定をブロック・パーミッション・テーブル（ＢＰＴ）に記述する。本発明のデバイスは、メディアに格納したデータ消去処理において、ＢＰＴを参照してＢＰＴに設定された各ブロックの消去可否を参照した後、消去可であるブロックのみの消去を実行する構成であるので、ＢＰＴ

格納ブロックを消去不可として設定したメディアについては、ＢＰＴの消去、書き換え替えが防止される。メディア内のＢＰＴを利用したファイルの書き込み、再生処理については後述する。

【００８３】

メディア（フラッシュメモリ搭載データ記録媒体）の製造時におけるブロック・パーミッション・テーブル（ＢＰＴ）の設定フローを、図１１および図１２に示す。ここでは、メディアとコマンド通信が行えるメディア作成器を通してメディア識別子（ＩＤ）の生成とＢＰＴの書き込みが連続動作で行われるものとする。

【００８４】

図１１は、相互認証処理機能を持たないメディア１のタイプにおけるメディア作成器が実行するブロック・パーミッション・テーブル（ＢＰＴ）の設定フローである。各処理について説明する。まず、まだ初期設定が行われていないメディアに対し、ＩＤ読み出しコマンドを送って（Ｓ３１）、あらかじめメディアに格納されたＩＤを受信（Ｓ３２）すると、そのＩＤをベースとしたＩＣＶ生成鍵Ｋｉｃｖ__bptを生成（Ｓ３３）する。ＩＣＶ生成鍵Ｋｉｃｖ__bptは、マスター鍵：ＭＫｉｃｖ_bptと、初期値：ＩＶｉｃｖ_bptと、ＢＰＴ識別子（ＩＤ）に基づいて生成する。具体的には、ＩＣＶ生成鍵Ｋｉｃｖ_ bpt＝ＤＥＳ（Ｅ，ＭＫｉｃｖ_bpt，ＩＤ＾ＩＶｉｃｖ_bpt）に基づいて生成される。式の意味は、ＢＰＴのＩＤと初期値ＩＶｉｃｖ_bptの排他論理和にマスター鍵：ＭＫｉｃｖ_bptによるＤＥＳモードでの暗号化処理を実行するという意味である。

【００８５】

次に、ＢＰＴの各フィールドに必要なパラメータを設定（Ｓ３４）し、各パラメータが設定されたＢＰＴに基づいてＩＣＶを生成（後述する図１４の構成を適用）し（Ｓ３５）、生成したＩＣＶをＢＰＴのＩＣＶフィールドに設定（Ｓ３６）する。このようにして構成されたブロック・パーミッション・テーブル（ＢＰＴ）をメディア１に書き込む（Ｓ３７）。なお、前述したようにＢＰＴの書き込みブロックは、ＢＰＴにおいて消去不可領域として設定されたブロックとする。

【００８６】

図１２は、相互認証処理機能を持つメディア２のタイプにおけるメディア作成器が実行するブロック・パーミッション・テーブル（ＢＰＴ）の設定フローである。各処理について説明する。まず、まだ初期設定が行われていないメディア２との相互認証処理およびセッション鍵の共有（これらの処理については、後述する図２２の処理を参照）を実行する。

【００８７】

相互認証および鍵共有処理が終了すると、メディア２に対しＩＤ読み出しコマンドを送って（Ｓ４１）、ＩＤを読み出し、ＩＤをベースとしたＩＣＶ生成鍵Ｋｉｃｖ＿bptを生成（Ｓ４２）する。ＩＣＶ生成鍵Ｋｉｃｖ＿bptは、マスター鍵：ＭＫｉｃｖ_bptと、初期値：ＩＶｉｃｖ_bptと、ＢＰＴ識別子（ＩＤ）に基づいて生成する。具体的には、ＩＣＶ生成鍵Ｋｉｃｖ_ｂｐｔ＝ＤＥＳ（Ｅ，ＭＫｉｃｖ_bpt，ＩＤ˙ＩＶｉｃｖ_bpt）に基づいて生成される。式の意味は、ＢＰＴのＩＤと初期値（ＩＶｉｃｖ_bpt）の排他論理和にマスター鍵：ＭＫｉｃｖ_bptによるＤＥＳモードでの暗号化処理を実行するという意味である。

【００８８】

次に、ＢＰＴの各フィールドに必要なパラメータを設定（Ｓ４５）し、各パラメータが設定されたＢＰＴに基づいてＩＣＶを生成（後述する図１４の構成を適用）し（Ｓ４６）、生成したＩＣＶをＢＰＴのＩＣＶフィールドに設定（Ｓ４７）する。このようにして構成されたブロック・パーミッション・テーブル（ＢＰＴ）をメディア１に書き込む（Ｓ４８）。なお、前述したようにＢＰＴの書き込みブロックは、ＢＰＴにおいて消去不可領域として設定されたブロックとする。

【００８９】

図１３にブロック・パーミッション・テーブル（ＢＰＴ）の具体的構成例を示す。図１３の（ａ）はメディア１、メディア２のフラッシュメモリのブロック構成であり、図１３（ｂ）は、ブロック・パーミッション・テーブル（ＢＰＴ）である。ブロック・パーミッション・テーブル（ＢＰＴ）は、フォーマット・バージョン、ＢＰＴＩＤ、ブロック数に続いて、各ブロックの消去可（１）、消去不可（０）が設定され、最後にＢＰＴの改竄チェック値（ＩＣＶ ｏｆ ＢＰＴ）が格納された構成を持つ。メモリのＢＰＴ格納ブロック（図１３の例ではブロック

＃２）は、ブロック・パーミッション・テーブル（ＢＰＴ）において消去不可領域として設定され、デバイスによる消去を防止し、ＢＰＴの書き替えが実行されない構成を持つ。

【００９０】

なお、図１３に示すブロック・パーミッション・テーブル（ＢＰＴ）の構成例は、各ブロックの消去可（１）、消去不可（０）のみが設定された構成であるが、消去処理のみのアクセス許可を設定する構成ではなく、読み取り（再生）許可、不許可を設定した構成としてもよい。例えば再生および消去不可（１１）、再生可、消去不可（１０）、再生不可、消去可（０１）、再生および消去可（００）とした設定が可能である。

【００９１】

なお、図２に示したようにメディア２ではメディア内に制御部２３１を持っており、ブロック・パーミッション・テーブル（ＢＰＴ）が設定済みかどうかの状態を保持することもでき、ＢＰＴが設定されている状態で、デバイスからＢＰＴの新たな書き込み命令が来たとしても、受け付けない構成として、ＢＰＴの再書き込みを防止する構成としてもよい。

【００９２】

なお、上述の例におけるＢＰＴ書き込みは、メディアとコマンド通信が行えるメディア作成器を通して実行する構成について説明したが、この他、メディアへのＢＰＴの書き込みは、単純なメモリライターで作成したＢＰＴを直接書き込む構成としてもよい。ただし、この場合も、メモリのＢＰＴ格納ブロックは、ブロック・パーミッション・テーブル（ＢＰＴ）において消去不可領域として設定する。

【００９３】

[改竄チェック値（ＩＣＶ）による改竄チェック]

次に、改竄チェック値（ＩＣＶ：Integrity Check Value）によるデータ改竄チェック処理について説明する。本発明の構成において、改竄チェック値（ＩＣＶ）は、データ記憶手段に格納されるコンテンツ、ブロック・パーミッション・テーブル、リボケーションリスト等に付加され、それぞれのデータ改竄チェック

処理に適用される。なお、コンテンツについての改竄チェック値は、セクタデータ単位に付加可能な構成である。コンテンツ、ブロック・パーミッション・テーブル、リボケーションリスト等に付加されたＩＣＶ処理の具体的形態については、後段で説明する。

【００９４】

ＤＥＳ暗号処理構成を用いた改竄チェック値（ＩＣＶ）生成例を図１４に示す。図１４の構成に示すように対象となる改竄チェックデータを構成するメッセージを８バイト単位に分割（以下、分割されたメッセージをＤ０、Ｄ１、Ｄ２、・・・、Ｄｎ－１とする）する。改竄チェックデータは、例えばコンテンツ自体であったり、上述したアクセス許可テーブルであるＢＰＴの構成データであったり、あるいはリボケーションリストの構成データである。

【００９５】

まず、初期値（Initial Value（以下、ＩＶとする））とＤ０を排他的論理和する（その結果をＩ１とする）。次に、Ｉ１をＤＥＳ暗号化部に入れ、改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖを用いて暗号化する（出力をＥ１とする）。続けて、Ｅ１およびＤ１を排他的論理和し、その出力Ｉ２をＤＥＳ暗号化部へ入れ、改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖを用いて暗号化する（出力Ｅ２）。以下、これを繰り返し、全てのメッセージに対して暗号化処理を施す。最後に出てきたＥＮをコンテンツチェック値ＩＣＶ’とする。

【００９６】

改竄のないことが保証された例えばコンテンツ生成時に生成した正当なＩＣＶと、新たにコンテンツに基づいて生成したＩＣＶ’とを比較して同一性が立証、すなわちＩＣＶ’＝ＩＣＶであれば入力メッセージ、例えばコンテンツ、ＢＰＴ、あるいはリボケーションリストに改竄のないことが保証され、ＩＣＶ’≠ＩＣＶであれば改竄があったと判定される。

【００９７】

ＩＣＶを使用したデータ改竄チェック処理フローを図１５に示す。まず、改竄チェックの対象データを抽出し（Ｓ１１）、抽出したデータに基づいて例えば図１４のＤＥＳ暗号処理構成によりＩＣＶ’を計算する（Ｓ１２）。計算の結果、

算出されたＩＣＶ’とデータ内に格納されたＩＣＶとを比較し（Ｓ１３）、一致した場合は、データの改竄が無く正当なデータであると判定（Ｓ１４からＳ１５）され、不一致の場合は、データの改竄があると判定（Ｓ１４からＳ１６）される。

【００９８】

リボケーションリストの改竄チェック用の改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｒｌは、予めデバイスのメモリインタフェース部３００のメモリ部３２１（図４参照）内に格納されたリボケーションリスト（Revocation List）のＩＣＶ鍵を生成するマスター鍵：ＭＫicv_rlと、リボケーションリスト（Revocation List）のＩＣＶ鍵を生成する時の初期値：ＩＶicv_rlと、リボケーションリストの属性情報中に含まれるリボケーションリスト・バージョン（Version）に基づいて生成する。具体的には、改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｒｌ＝ＤＥＳ（Ｅ，ＭＫicv_rl，Version˚ＩＶicv_rl）に基づいて生成される。前記式の意味は、バージョン（Version）と初期値（ＩＶicv_rl）の排他論理和にマスター鍵：ＭＫicv_rlによるＤＥＳモードでの暗号化処理を実行するという意味である。リボケーションリストの改竄チェック値は、このようにして生成されたＩＣＶ生成鍵Ｋｉｃｖ_rlを適用して初期値ＩＶrl（メモリ部３２１に格納）を用いて図１５に示すＩＣＶ生成構成によって実行される。

【００９９】

また、ブロック・パーミッション・テーブル（ＢＰＴ）の改竄チェック用の改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｂｐｔは、予めデバイスのメモリインタフェース部３００のメモリ部３２１（図４参照）内に格納されたＢＰＴのＩＣＶ鍵を生成するマスター鍵：ＭＫicv_bptと、ＢＰＴのＩＣＶ鍵を生成する時の初期値：ＩＶicv_bptと、ＢＰＴの属性情報中に含まれるＢＰＴ識別子（ＩＤ）に基づいて生成する。具体的には、改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｂｐｔ＝ＤＥＳ（Ｅ，ＭＫicv_bpt，ＩＤ˚ＩＶicv_bpt）に基づいて生成される。前記式の意味は、ＢＰＴのＩＤと初期値（ＩＶicv_bpt）の排他論理和にマスター鍵：ＭＫicv_bptによるＤＥＳモードでの暗号化処理を実行するという意味である。ブロック・パーミッション・テーブル（ＢＰＴ）の改竄チェック値は、

このようにして生成されたＩＣＶ生成鍵Ｋｉｃｖ_bptを適用して初期値ＩＶbpt（メモリ部３２１に格納）を用いて図１５に示すＩＣＶ生成構成によって実行される。なお、ＢＰＴの付帯情報として格納されるＩＣＶは、ＢＰＴ内のデータとＢＰＴを格納したメディアの識別子（ＩＤ）を含むデータに基づいて生成される。従って、ＢＰＴのＩＣＶチェックは、ＢＰＴのデータ改竄の有無のみならず、メディア固有の正当なＢＰＴ、すなわち他のメディアにコピーされたＢＰＴでないことを検証する機能も兼ね備える。

【０１００】

また、コンテンツのセクタ単位の改竄チェック用の改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｃｏｎｔは、コンテンツのヘッダ（セキュリテイ・ヘッダ）中に暗号化されて格納されており、必要に応じてメモリインタフェースの暗号処理部３２０（図４参照）において、また、メディア２との相互認証後に実行されるメディア２のコントローラ２３１で実行されるＤＥＳ－ＣＢＣモードによる復号処理によって取得される。これらの処理についてはフローを用いた説明中で詳細に説明する。

【０１０１】

このようなデータ改竄チェックの結果、例えばリボケーションリストの改竄が明らかになれば、リボケーションリストの参照処理に基づくコンテンツの再生等の処理を禁止し、また、アクセス許可テーブルであるＢＰＴに改竄があると判定されれば、ＢＰＴに基づくメディアのデータに対するアクセスを禁止する処理を実行する。これらの処理については、後段で詳細に説明する。

【０１０２】

[データ読み出し、書き込み処理]

以下、本発明のデータ処理装置において、デバイスがメディアからのデータ読み出しを行なう場合の処理、およびデバイスがメディアに対してデータを格納する場合に実行される処理について説明する。

【０１０３】

（デバイス起動時処理）

まず、デバイスを起動させた場合における処理を図１６を用いて説明する。図

１６は、左側に図２におけるデバイス２００の制御部２０５の処理、右側にメモリインタフェース部３００の処理を示したものである。処理スタート時点でのメモリインタフェース部３００のステータスレジスタの状態は、ビジーフラグ：０（待機）、リボケーションリストセットフラグ：０（未セット）である。

【０１０４】

まず、デバイスが起動すると、制御部は、内部メモリのファイル割り当てテーブル呼び出しコマンドをメモリインタフェース部に送信（Ｓ１０１）する。メモリインタフェース部は、デバイスの内部メモリに対してファイル割り当てテーブルの読み出しコマンドを送信（Ｓ１０２）して、ファイル割り当てテーブルを内部メモリから受信し、制御部に送信（Ｓ１０３）する。

【０１０５】

なお、ファイル割り当てテーブルは、デバイスのアクセス可能な内部メモリ、外部メモリに格納されたデータ、例えば様々なコンテンツ、あるいはリボケーションリスト等、各種データファイルをディレクトリ管理するテーブルであり、例えば図１７に示すように、ディレクトリ、ファイル名、格納セクタが対応付けられた構成を持つ。デバイスは、ファイル割り当てテーブルに基づいて、様々なファイルのアクセスを行なう。

【０１０６】

制御部は、内部メモリに格納されたデータに対応するファイル割り当てテーブルを受信（Ｓ１０４）すると、テーブルに基づいてリボケーションリストの読み出し処理を実行（Ｓ１０５）し、リボケーションリストのセットコマンドと、リボケーションリストをメモリインタフェースに送信（Ｓ１０６）する。リボケーションリストのセット処理は、リボケーションリストが有効である場合にのみ実行され、リストがセットされると、メディアからのコンテンツ読み出し処理等、コンテンツ処理の際、リボケーションリストにリストアップされたコンテンツまたはメディア識別子との比較処理を実行する。これらの処理については後述する。

【０１０７】

リボケーションリストのセットコマンドと、リボケーションリストを制御部か

ら受信（Ｓ１０７）すると、メモリインタフェースは、ステータスレジスタのビジーフラグを１（ビジー）にセット（Ｓ１０８）し、リボケーションリストの改竄チェック用の改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ＿ｒｌを生成（Ｓ１０９）する。

【０１０８】

リボケーションリストの改竄チェック用の改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｒｌは、予めデバイス内に格納されたリボケーションリスト（Revocation List）のＩＣＶ鍵を生成するマスター鍵：ＭＫicv_rlと、リボケーションリスト（Revocation List）のＩＣＶ鍵を生成する時の初期値：ＩＶicv_rlと、リボケーションリストの属性情報中に含まれるリボケーションリスト・バージョン（Version）に基づいて生成する。具体的には、改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｒｌ＝ＤＥＳ（Ｅ，ＭＫicv_rl，Version＾ＩＶicv_rl）に基づいて生成される。式の意味は、バージョン（Version）と初期値（ＩＶicv_rl）の排他論理和にマスター鍵：ＭＫicv_rlによるＤＥＳモードでの暗号化処理を実行するという意味である。

【０１０９】

次にメモリインタフェースは生成した改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｒｌを用いてリボケーションリストのＩＣＶ’を生成し、予めリボケーションリスト内に格納された正しいＩＣＶとの照合処理（ＩＣＶ’＝ＩＣＶ？）を実行（Ｓ１１０）する。なお、ＩＣＶ’の生成処理は、前述の図１４で説明したＤＥＳモードに基づいて、初期値ＩＶrlを用い、生成した改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｒｌを適用した処理によって行われる。

【０１１０】

ＩＣＶ’＝ＩＣＶである場合（Ｓ１１１でＹｅｓ）は、リボケーションリストが改竄のない正当なものであると判定され、コンテンツの読み出し処理等の際に参照可能な状態にセットし、リボケーションリストセットフラグを１（セット）にセット（Ｓ１１２）する。リボケーションリストはメモリインタフェース内のメモリ（例えばメモリ部３２１（図４参照））に格納され、例えば、送受信制御部３０６が制御部２０５（図２参照）からメディア認識コマンドを受信するとセ

ットされたリボケーションリストのメデイア識別子と、デバイスに装着したメディアのメディア識別子との照合が実行され、また、送受信制御部３０６が制御部２０５からコンテンツの読み出し処理に伴うヘッダセットコマンドを受信するとセットされたリボケーションリストのコンテンツ識別子と、読み出し対象コンテンツのコンテンツ識別子との照合が実行される。

【０１１１】

このように、リボケーションリストは、外部メモリ等に直接アクセスするメモリインタフェースにセットアップされ、セットアップ後は、メデイアの装着時、コンテンツの再生時においてメモリインタフェースにおいて継続的に利用可能な構成とされ、コンテンツの利用時に繰り返し内部メモリから読み出すなどの処理が不要となり処理が効率的に実行される。

【０１１２】

図１６のフローの説明を続ける。ＩＣＶ’≠ＩＣＶである場合（Ｓ１１１でＮｏ）は、リボケーションリストに改竄ありと判定され、リストの参照処理に基づくコンテンツ処理を禁止し処理を終了する。以上の処理の終了により、ビジーフラグは０にセットされる。

【０１１３】

一方、制御部側は、ステータス読み出しコマンドをメモリインタフェースに送信（Ｓ１１４）し、ビジーフラグが０となったことを条件（Ｓ１１５）としてリボケーションリストセットフラグを保存（Ｓ１１６）する。保存されるリボケーションセットフラグは、リストの改竄が無いと判定された場合は、リストが有効にセットされたことを示す１、その他の場合は０となる。

【０１１４】

（メディア認識時処理）

次に、デバイスにメディアが装着された場合のメディアの有効性確認等、メディア認識時に実行する処理について説明する。前述したようにメディアには、デバイスとの相互認証処理を実行しないタイプのメディア１と、デバイスとの相互認証処理を実行するタイプのメディア２とがある。デバイスは、それぞれのタイプがデバイスに装着されると、メディアを利用したコンテンツ処理を実行してよ

いか否か、具体的にはリボケーションリストに不正メディアとしての登録がないかを確認する処理を実行し、装着メディアがリボケーションリストにリストアップされておらず、有効に利用可能なメディアであることが確認されたことを条件として、メディアに格納されたアクセス許可テーブルであるＢＰＴ（Block Permission Table）をメモリインタフェースにセットし、ＢＰＴを参照したメモリアクセスを可能とする処理を実行する。

【０１１５】

まず、メディア１が装着された場合のメディア確認処理について図１８、図１９を用いて説明する。

【０１１６】

図１８、図１９においても左側に図２におけるデバイス２００の制御部２０５の処理、右側にメモリインタフェース部３００の処理を示している。当フロー開始時点で、メモリインタフェース部３００のステータスレジスタの状態は、ビジーフラグ：０（待機）、メディア１有効フラグ：０（無効）、メディア１セットフラグ：０（未セット）の状態である。

【０１１７】

まず、制御部は、デバイスに装着されたメディアがメディア１であることを認識する（Ｓ２０１）。メディア識別は予め設定されたメディア形状に基づく機械的情報あるいはデバイス、メディア間の通信情報に基づいて行われる。制御部がメディア１であることを認識すると制御部は、メディア１認識コマンドをメモリインタフェースに送信する（Ｓ２０２）。

【０１１８】

メモリインタフェースは、制御部からのメディア１認識コマンドを受信（Ｓ２０３）すると、ステータスレジスタのビジーフラグを１（ビジー）に設定し（Ｓ２０４）、メディア１に対してメディア１の識別子（ＩＤ）の読み出しコマンドを送信（Ｓ２０５）し、受信（Ｓ２０６）する。さらに、受信したメディア１のＩＤと、既にセットされているリボケーションリスト中のリボーク（排除）メディア１のリストとの比較照合を実行（Ｓ２０７）する。リボケーションリストは、先の図１６の起動時フローにおいて説明したように、起動時にメモリインタフ

ェースにセットアップされ、セットアップ後は、メデイアの装着時、コンテンツの再生時においてメモリインタフェースにおいて継続的に利用可能となる。

【０１１９】

受信ＩＤと一致するＩＤがリスト中に存在しなかった場合は、装着メディア１はリボーク対象メディアではなく、有効に利用可能なメデイアであると判定（Ｓ２０８においてＮｏ）し、ステータスレジスタのメディア１有効フラグを１（有効）にセット（Ｓ２０９）し、ビジーフラグを０（待機）にセット（Ｓ２１０）する。受信ＩＤと一致するＩＤがリボケーションリスト中にあった場合（Ｓ２０８においてＹｅｓ）は、装着メディア１はリボーク対象メディアであり、有効に利用できないと判定し、ステップＳ２０９の有効フラグの有効化処理を実行せずステップＳ２１０でビジーフラグを０（待機）にセットして処理を終了する。

【０１２０】

一方、制御部は、ステップＳ２１１において、ステータス読み出しコマンドをメモリインタフェースに送信し、ビジーフラグが０（待機）になったことを確認（Ｓ２１２）の後、メディアフラグ状態を確認して有効（フラグ：１）である場合（Ｓ２１３でＹｅｓ）にのみ処理を続行し、無効（フラグ：０）である場合（Ｓ２１３でＮｏ）は、処理を終了する。

【０１２１】

次に、図１９に進み、制御部は、メディア１に関するファイル割り当てテーブル呼び出しコマンドをメモリインタフェースに送信（Ｓ２２１）し、メモリインタフェースは、ファイル割り当てテーブルの格納されたセクタ読み出しコマンドをメディア１に送信（Ｓ２２２）し、ファイル割り当てテーブルをメディア１から受信し、制御部に送信（Ｓ２２３）する。

【０１２２】

制御部は、メディア１に格納されたデータに対応するファイル割り当てテーブルを受信（Ｓ２２４）すると、テーブルに基づいてブロック・パーミッション・テーブル（ＢＰＴ）の読み出し処理を実行（Ｓ２２５）し、ＢＰＴのセットコマンドと、ＢＰＴをメモリインタフェースに送信（Ｓ２２６）する。ＢＰＴのセット処理は、ＢＰＴが有効である場合にのみ実行され、ＢＰＴがセットされると、

メディアからのコンテンツ書き込み処理等、コンテンツ処理の際、ＢＰＴを参照してブロック毎の消去が可能か否かを判定する。実際のＢＰＴを参照したデータ書き込み処理については、後段で説明する。

【０１２３】

ブロック・パーミッション・テーブル（ＢＰＴ）のセットコマンドと、ＢＰＴを制御部から受信（Ｓ２２７）すると、メモリインタフェースは、ステータスレジスタのビジーフラグを１（ビジー）にセット（Ｓ２２８）し、ＢＰＴの改竄チェック用の改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ＿ｂｐｔを生成（Ｓ２２９）する。

【０１２４】

ＢＰＴの改竄チェック用の改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｂｐｔは、予めデバイス内に格納されたＢＰＴのＩＣＶ鍵を生成するマスター鍵：ＭＫicv_bptと、ＢＰＴのＩＣＶ鍵を生成する時の初期値：ＩＶicv_bptと、メディアＩＤに基づいて生成する。具体的には、改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｂｐｔ＝ＤＥＳ（Ｅ，ＭＫicv_bpt，メディア１ＩＤ＾ＩＶicv_bpt）に基づいて生成される。式の意味は、メディア１ＩＤと初期値（ＩＶicv_bpt）の排他論理和にマスター鍵：ＭＫicv_bptによるＤＥＳモードでの暗号化処理を実行するという意味である。

【０１２５】

次にメモリインタフェースは生成した改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_bptを用いてＢＰＴのＩＣＶ’を生成し、予めＢＰＴ内に格納された正しいＩＣＶ値との照合処理（ＩＣＶ’＝ＩＣＶ？）を実行（Ｓ２３０）する。なお、ＩＣＶ’の生成処理は、前述の図１４で説明したＤＥＳモードに基づいて、初期値ＩＶbptを用い、生成した改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_bptを適用した処理によって行われる。なお、ＢＰＴの付帯情報として格納されたＩＣＶは、メディアＩＤを含むデータに基づいて生成されており、ＩＣＶのチェックは、ＢＰＴのデータ改竄の有無のみならず、メディア固有の正当なＢＰＴ、すなわち他のメディアにコピーされたＢＰＴでないことの検証も兼ね備える機能を持つ。

【０１２６】

ＩＣＶ’＝ＩＣＶである場合（Ｓ２３１でＹｅｓ）は、ＢＰＴが正当なメディアに格納された改竄のない正当なものであると判定され、コンテンツ処理等の際に参照可能な状態にセットし、メディア１セットフラグを１（セット）にセット（Ｓ２３２）する。ＩＣＶ’≠ＩＣＶである場合（Ｓ２３１でＮｏ）は、ＢＰＴに改竄ありと判定され、ＢＰＴの参照処理に基づくコンテンツ処理を禁止し処理を終了する。以上の処理の終了により、ビジーフラグは０にセット（Ｓ２３３）される。

【０１２７】

一方、制御部側は、ステータス読み出しコマンドをメモリインタフェースに送信（Ｓ２３４）し、ビジーフラグが０となったことを条件（Ｓ２３５でＹｅｓ）としてメディア１セットフラグを保存（Ｓ２３６）する。保存されるメディア１セットフラグは、ＢＰＴの改竄が無いと判定された場合は、メディア１が有効にセットされたことを示す１、その他の場合は０となる。

【０１２８】

次にメディア２がデバイスに装着された際のメディア２確認処理について、図２０、図２１を用いて説明する。メディア２は、図２を用いて説明したように、デバイスとの相互認証を実行するメディアである。

【０１２９】

図２０のステップＳ３０１からＳ３０４のステップは、メディア１の確認処理におけるステップＳ２０１～Ｓ２０４と同様であるので説明を省略する。

【０１３０】

ステップＳ３０５において、メモリインタフェースは、メディア２との相互認証処理を実行する。

【０１３１】

図２２に、共通鍵暗号方式を用いた相互認証方法（ISO/IEC 9798-2）の処理シーケンスを示す。図２２においては、共通鍵暗号方式としてＤＥＳを用いているが、共通鍵暗号方式であれば他の方式も可能である。図２２において、まず、Ｂが６４ビットの乱数Ｒｂを生成し、Ｒｂおよび自己のＩＤであるＩＤ（ｂ）をＡに送信する。これを受信したＡは、新たに６４ビットの乱数Ｒａを生成し、Ｒａ

、Ｒｂ、ＩＤ（ｂ）の順に、ＤＥＳのＣＢＣモードで鍵Ｋａｂを用いてデータを暗号化し、Ｂに返送する。なお、鍵Ｋａｂは、ＡおよびＢに共通の秘密鍵、認証鍵である。ＤＥＳのＣＢＣモードを用いた鍵Ｋａｂによる暗号化処理は、例えばＤＥＳを用いた処理においては、初期値とＲａとを排他的論理和し、ＤＥＳ暗号化部において、鍵Ｋａｂを用いて暗号化し、暗号文Ｅ１を生成し、続けて暗号文Ｅ１とＲｂとを排他的論理和し、ＤＥＳ暗号化部において、鍵Ｋａｂを用いて暗号化し、暗号文Ｅ２を生成し、さらに、暗号文Ｅ２とＩＤ（ｂ）とを排他的論理和し、ＤＥＳ暗号化部において、鍵Ｋａｂを用いて暗号化して生成した暗号文Ｅ３とによって送信データ（Token-AB）を生成する。

【０１３２】

これを受信したＢは、受信データを、やはり共通の秘密鍵としてそれぞれの記録素子内に格納する鍵Ｋａｂ（認証鍵）で復号化する。受信データの復号化方法は、まず、暗号文Ｅ１を認証鍵Ｋａｂで復号化し、初期値と排他的論理和し乱数Ｒａを得る。次に、暗号文Ｅ２を認証鍵Ｋａｂで復号化し、その結果とＥ１を排他的論理和し、Ｒｂを得る。最後に、暗号文Ｅ３を認証キーＫａｂで復号化し、その結果とＥ２を排他的論理和し、ＩＤ(ｂ)を得る。こうして得られたＲａ、Ｒｂ、ＩＤ(ｂ)のうち、ＲｂおよびＩＤ(ｂ)が、Ｂが送信したものと一致するか検証する。この検証に通った場合、ＢはＡを正当なものとして認証する。

【０１３３】

次にＢは、認証後に使用するセッションキー（Ｋｓｅｓ）を乱数によって生成する。そして、Ｒｂ、Ｒａ、Ｋｓｅｓの順に、ＤＥＳのＣＢＣモードで認証キーＫａｂを用いて暗号化し、Ａに返送する。

【０１３４】

これを受信したＡは、受信データを認証キーＫａｋｅで復号化する。受信データの復号化方法は、Ｂの復号化処理と同様である。こうして得られたＲｂ、Ｒａ、Ｋｓｅｓの内、ＲｂおよびＲａが、Ａが送信したものと一致するか検証する。この検証に通った場合、ＡはＢを正当なものとして認証する。互いに相手を認証した後には、セッションキーＫｓｅｓは、認証後の秘密通信のための共通鍵として利用される。

【０１３５】

なお、受信データの検証の際に、不正、不一致が見つかった場合には、相互認証が失敗したものとして、その後の相互間のデータ通信処理が禁止される。

【０１３６】

図２３、図２４に本発明のデバイスとメディア間における相互認証、鍵（セッション鍵）共有処理フローを示す。図２３、図２４において、左側がデバイスのメモリインタフェース、右側がメディア２のコントローラにおける処理である。

【０１３７】

まず、メディア２コントローラが乱数Ｒａを生成（Ｓ４０１）し、Ｒａおよび自己のＩＤであるメディア２ＩＤをデバイスメモリインタフェースに送信（Ｓ４０２）する。これを受信（Ｓ４０３）したデバイスメモリインタフェースは、受信したメディア２ＩＤと、初期値（ＩＶ__ake）の排他論理和に自己の所有する認証鍵生成用マスター鍵：ＭＫａｋｅを適用しＤＥＳ暗号化処理を行なって認証鍵Ｋａｋｅを生成（Ｓ４０４）する。さらに、デバイスメモリインタフェースは、新たに乱数Ｒｂを生成（Ｓ４０５）し、初期値ＩＶ__ａｕｔｈとＲｂとを排他的論理和し、鍵Ｋａｋｅを用いて暗号化し、暗号文Ｅ１を生成し、続けて暗号文Ｅ１とＲａとを排他的論理和し、鍵Ｋａｋｅを用いて暗号化して暗号文Ｅ２を生成し、さらに、暗号文Ｅ２とメディア２ＩＤとを排他的論理和し、鍵Ｋａｋｅを用いて暗号化して暗号文Ｅ３を生成し（Ｓ４０６）、生成したデータＥ１||Ｅ２||Ｅ３をメディア２コントローラに送信（Ｓ４０７）する。［||］は、データの結合を意味する。

【０１３８】

これを受信（Ｓ４０８）したメディア２コントローラは、受信データを、認証鍵Ｋａｋｅで復号化（Ｓ４０９）する。受信データの復号化方法は、まず、暗号文Ｅ１を認証鍵Ｋａｋｅで復号化し、初期値と排他的論理和し乱数Ｒｂ’を得る。次に、暗号文Ｅ２を認証鍵Ｋａｋｅで復号し、その結果とＥ１を排他的論理和し、Ｒａ’を得る。最後に、暗号文Ｅ３を認証鍵Ｋａｋｅで復号し、その結果とＥ２を排他的論理和し、メディア２ＩＤ’を得る。こうして得られたＲａ’、Ｒｂ’、メディア２ＩＤ’のうち、Ｒａ’およびメディア２ＩＤ’が、メディア２

が送信したものと一致するか検証（Ｓ４１０，Ｓ４１１）する。この検証に通った場合、メディア２はデバイスを正当なものとして認証する。Ｒａ’およびメディア２ＩＤ’が、送信データと不一致であったときは、相互認証が失敗（Ｓ４１３）したものとし、その後のデータ通信を中止する。

【０１３９】

次にメディア２コントローラは、認証後に使用するセッションキー（Ｋｓｅｓ）としての乱数を生成（Ｓ４１２）する。次に、図２４のステップＳ４２１において、Ｒａ、Ｒｂ、Ｋｓｅｓの順に、ＤＥＳのＣＢＣモードで認証鍵Ｋａｋｅを用いて暗号化し、デバイスメモリインタフェースに送信（Ｓ４２２）する。

【０１４０】

これを受信（Ｓ４２３）したデバイスメモリインタフェースは、受信データを認証鍵Ｋａｋｅで復号（Ｓ４２４）する。こうして得られたＲａ”、Ｒｂ”、Ｋｓｅｓの内、Ｒａ”およびＲｂ”が、デバイスが送信したものと一致するか検証（Ｓ４２５，Ｓ４２６）する。この検証に通った場合、デバイスはメディア２を正当なものとして認証（Ｓ４２７）する。互いに相手を認証した後には、セッションキーＫｓｅｓを共有（Ｓ４２９）し、認証後の秘密通信のための共通鍵として利用される。Ｒａ”およびＲｂ”が、送信データと不一致であったときは、相互認証が失敗（Ｓ４２８）したものとし、その後のデータ通信を中止する。

【０１４１】

図２０に戻り、メディア２の認識処理について説明を続ける。ステップＳ３０５において上述の相互認証、鍵共有処理が実行され、ステップＳ３０６で相互認証が成功したことが確認されると、相互認証処理時に受信したメディア２のＩＤと、既にセットされているリボケーションリスト中のリボーク（排除）メディア２のリストとの比較照合を実行（Ｓ３０７）する。

【０１４２】

受信ＩＤと一致するＩＤがリスト中に存在しなかった場合は、装着メディア２はリボーク対象メディアではなく、有効に利用可能なメデイアであると判定（Ｓ３０８においてＮｏ）し、ステータスレジスタのメディア２有効フラグを１（有効）にセット（Ｓ３０９）し、ビジーフラグを０（待機）にセット（Ｓ３１０）

する。受信ＩＤと一致するＩＤがリボケーションリスト中にあった場合（Ｓ３０８においてＹｅｓ）は、装着メディア２はリボーク対象メディアであり、有効に利用できないと判定し、ステップＳ３０９の有効フラグの有効化処理を実行せずステップＳ３１０でビジーフラグを０（待機）にセットして処理を終了する。

【０１４３】

一方、制御部は、ステップＳ３１１において、ステータス読み出しコマンドをメモリインタフェースに送信し、ビジーフラグが０（待機）になったことを確認（Ｓ３１２）の後、メディアフラグ状態を確認して有効（フラグ：１）である場合（Ｓ３１３でＹｅｓ）にのみ処理を続行し、無効（フラグ：０）である場合（Ｓ３１３でＮｏ）は、処理を終了する。

【０１４４】

次に、図２１に進み、制御部は、メディア２に関するファイル割り当てテーブル呼び出しコマンドをメモリインタフェースに送信（Ｓ３２１）し、メモリインタフェースは、ファイル割り当てテーブルの格納されたセクタ読み出しコマンドをメディア２に送信（Ｓ３２２）し、ファイル割り当てテーブルをメディア２から受信し、制御部に送信（Ｓ３２３）する。

【０１４５】

制御部は、メディア２に格納されたデータに対応するファイル割り当てテーブルを受信（Ｓ３２４）すると、テーブルに基づいてブロック・パーミッション・テーブル（ＢＰＴ）の読み出し処理を実行（Ｓ３２５）し、ＢＰＴのセットコマンドと、ＢＰＴをメモリインタフェースに送信（Ｓ３２６）する。ＢＰＴのセット処理は、ＢＰＴが有効である場合にのみ実行され、ＢＰＴがセットされると、メディアからのコンテンツ書き込み処理等、コンテンツ処理の際、ＢＰＴを参照してブロック毎の消去が可能か否かを判定する。実際のＢＰＴを参照したデータ書き込み処理については、後段で説明する。

【０１４６】

ブロック・パーミッション・テーブル（ＢＰＴ）のセットコマンドと、ＢＰＴを制御部から受信（Ｓ３２７）すると、メモリインタフェースは、ステータスレジスタのビジーフラグを１（ビジー）にセット（Ｓ３２８）し、ＢＰＴの改竄チ

エック用の改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ＿ｂｐｔを生成（Ｓ３２９）する。

【０１４７】

ＢＰＴの改竄チェック用の改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｂｐｔは、予めデバイス内に格納されたＢＰＴのＩＣＶ鍵を生成するマスター鍵：ＭＫicv_bptと、ＢＰＴのＩＣＶ鍵を生成する時の初期値：ＩＶicv_bptと、メディア２ＩＤに基づいて生成する。具体的には、改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｂｐｔ＝ＤＥＳ（Ｅ，ＭＫicv_bpt，・ディア２ＩＤ˙ＩＶicv_bpt）に基づいて生成される。式の意味は、メディア２ＩＤと初期値（ＩＶicv_bpt）の排他論理和にマスター鍵：ＭＫicv_bptによるＤＥＳモードでの暗号化処理を実行するという意味である。

【０１４８】

次にメモリインタフェースは生成した改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_bptとＩＶbptを用いてＢＰＴのＩＣＶ’を生成し、予めＢＰＴ内に格納された正しいＩＣＶ値との照合処理（ＩＣＶ’＝ＩＣＶ？）を実行（Ｓ３３０）する。なお、ＩＣＶ’の生成処理は、前述の図１４で説明したＤＥＳモードに基づいて、初期値ＩＶbptを用い、生成した改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_bptを適用した処理によって行われる。なお、ＢＰＴの付帯情報として格納されたＩＣＶは、メディア２ＩＤを含むデータに基づいて生成されており、ＩＣＶのチェックは、ＢＰＴのデータ改竄の有無のみならず、メディア固有の正当なＢＰＴ、すなわち他のメディアにコピーされたＢＰＴでないことの検証も兼ね備える機能を持つ。

【０１４９】

ＩＣＶ’＝ＩＣＶである場合（Ｓ３３１でＹｅｓ）は、ＢＰＴが正当なメディアに格納された改竄のない正当なものであると判定され、コンテンツ処理等の際に参照可能な状態にセットし、メディア２セットフラグを１（セット）にセット（Ｓ３３２）する。ＩＣＶ’≠ＩＣＶである場合（Ｓ３３１でＮｏ）は、ＢＰＴに改竄ありと判定され、ＢＰＴの参照処理に基づくコンテンツ処理を禁止し処理を終了する。以上の処理の終了により、ビジーフラグは０にセット（Ｓ３３３）

される。

【０１５０】

一方、制御部側は、ステータス読み出しコマンドをメモリインタフェースに送信（Ｓ３３４）し、ビジーフラグが０となったことを条件（Ｓ３３５でＹｅｓ）としてメディア２セットフラグを保存（Ｓ３３６）する。保存されるメディア２セットフラグは、ＢＰＴの改竄が無いと判定された場合は、メディア２が有効にセットされたことを示す１、その他の場合は０となる。

【０１５１】

（データファイル読み出し処理）

次に、データファイルの読み出し処理について図２５のフローを用いて説明する。データファイルには、音楽データ、画像データ等のコンテンツデータファイル、さらに前述したリボケーションリストも含まれる。図２５に示すフローは、内部メモリ、外部メモリ（メディア１、メディア２）のいずれかに格納されたデータファイルの読み出しに共通な処理フローである。図２５において、左側がデバイスの制御部、右側がデバイスのメモリインタフェースの処理である。

【０１５２】

まず、制御部は、ファイル割り当てテーブル（図１７参照）から読み出し対象データのセクタアドレス（Ｓ（１）～Ｓ（ｋ））を取得（Ｓ５０１）し、メモリインタフェースに取得したセクタＳ（ｉ）読み出しコマンドを順次送信（Ｓ５０２，Ｓ５０３）する。メモリインタフェースは、セクタＳ（ｉ）読み出しコマンドを受信（Ｓ５０４）すると、ビジーフラグを１（ビジー）に設定（Ｓ５０５）し、受信セクタＳ（ｉ）が内部メモリか、外部メモリであるかを判定（Ｓ５０６）し、外部メモリである場合は、メディア１かメディア２のセットフラグが１（メディアが有効にセットされていることを示す）であるかを判定（Ｓ５０７）し、セットフラグが１である場合には、さらにブロックパーミッション・テーブル（ＢＰＴ）を参照して、ＢＰＴが読み出し対象であるセクタＳ（ｉ）を読み出し許可対象ブロックとして設定しているかを判定（Ｓ５０８）する。ＢＰＴに読み出し許可ブロックの設定がある場合には、外部メモリから該当セクタのデータを読み出す（Ｓ５０９）。

【０１５３】

なお、読み出し対象データがＢＰＴによる管理のなされていない内部メモリ内のデータである場合は、ステップＳ５０７，Ｓ５０８はスキップする。ステップＳ５０７、Ｓ５０８の判定がＮｏである場合、すなわちセクタＳ（ｉ）を格納したメディアのセットフラグが１でない場合、または、ＢＰＴにセクタＳ（ｉ）の読み出し許可が設定されていない場合には、ステップＳ５１３に進み、読み出しエラーとして読み出し成功フラグが０にセットされる。

【０１５４】

ステップＳ５０６～Ｓ５０８の判定ブロックにおいて、対象セクタＳ（ｉ）の読み出しが実行可と判定されると、メモリから該当セクタが読み出され、セクタに対応して設定されている冗長部の誤り訂正符号に基づく誤り訂正処理が実行（Ｓ５１０）され、誤り訂正が成功した（Ｓ５１１）ことを確認し、読み出し成功フラグを１（成功）にセットし、読み出し結果をバッファに格納（Ｓ５１２）し、ビジーフラグを０（待機）に設定（Ｓ５１３）する。誤り訂正に失敗した場合は、読み出し成功フラグを０（失敗）に設定（Ｓ５１３）して処理を終了する。

【０１５５】

また、制御部は、ステップＳ５１５～Ｓ５２０において、メモリインタフェースのステータスを読み出して、ビジーフラグが０の状態において、読み出し成功フラグが１であることを条件として読み出しデータをバッファから取り出して保存し、アドレスを順次インクリメントして、データを順次バッファから取り出して保存する処理を繰り返し実行し、すべての読み出し対象セクタを保存した後、全読み出しセクタデータからファイルを構成して処理を終了する。

【０１５６】

（ファイル書き込み処理）

次に、データファイルの書き込み処理について図２６のフローを用いて説明する。図２６に示すフローは、内部メモリ、外部メモリ（メディア１、メディア２）のいずれかにファイルを書き込む際の共通処理フローである。図２６において、左側がデバイスの制御部、右側がデバイスのメモリインタフェースの処理である。

【０１５７】

まず、制御部は、書き込み対象ファイルをセクタに分割する。分割されたデータをＤ（１）～Ｄ（ｋ）とする。制御部は、次に各データＤ（ｉ）の書き込みセクタＳ（ｉ）を設定して、メモリインタフェースにセクタＳ（ｉ）書き込みコマンドと、データＤ（ｉ）を順次送信（Ｓ６０２～Ｓ６０４）する。メモリインタフェースは、セクタＳ（ｉ）書き込みコマンドを受信（Ｓ６０５）すると、ビジーフラグを１（ビジー）に設定（Ｓ６０６）し、受信セクタＳ（ｉ）が内部メモリか、外部メモリであるかを判定（Ｓ６０７）し、外部メモリである場合は、メディア１かメディア２のセットフラグが１（メディアが有効にセットされていることを示す）であるかを判定（Ｓ６０８）し、セットフラグが１である場合には、さらにブロックパーミッション・テーブル（ＢＰＴ）を参照して、ＢＰＴが書き込み対象であるセクタＳ（ｉ）を書き込み許可対象ブロックとして設定しているかを判定（Ｓ６０９）する。ＢＰＴに書き込み許可ブロックの設定がある場合には、セクタに対応して設定する誤り訂正符号を生成（Ｓ６１０）し、セクタＳ（ｉ）にデータＤ（ｉ）と誤り訂正符号を持つ冗長部を書き込み、書き込み成功フラグを１（成功）にセットし、ビジーフラグを０（待機）に設定（Ｓ６１４）する。

【０１５８】

なお、書き込み対象データがＢＰＴによる管理のなされていない内部メモリ内への書き込み処理である場合は、ステップＳ６０８，Ｓ６０９はスキップする。ステップＳ６０８、Ｓ６０９の判定がＮｏである場合、すなわちメディアのセットフラグが１でない場合、または、ＢＰＴにセクタＳ（ｉ）の書き込み許可が設定されていない場合には、ステップＳ６１３に進み、書き込みエラーとして書き込み成功フラグを０にセットする。

【０１５９】

また、制御部は、ステップＳ６１６～Ｓ６２０において、メモリインタフェースのステータスを読み出して、ビジーフラグが０の状態において、書き込み成功フラグが１であることを条件としてアドレスを順次インクリメントして、書き込みデータを順次メモリインタフェースに送信する。すべての処理が終了すると、

ファイル割り当てテーブルの更新処理を実行（Ｓ６２１）し、更新したファイル割り当てテーブルを更新コマンドとともにメモリインタフェースに送信（Ｓ６２２）し、メモリインタフェースはコマンドに従ってファイル割り当てテーブルの書き込み処理を実行（Ｓ６２３）する。

【０１６０】

［セクタ位置に応じた暗号化鍵を適用した暗号化処理］

次に、セクタ位置に応じた暗号化鍵を適用した暗号化処理について説明する。著作権などを保護するためにコンテンツ部に対する暗号化を行う場合があるが、コンテンツ部全体に対して一つの暗号化鍵を使って暗号化すると、同一の鍵の元での大量の暗号文が発生し、攻撃が容易となってしまう危険性がある。通常はコンテンツ部をできるだけ分割し、それぞれを異なる鍵で暗号化する方が望ましいと言える。本システムでのコンテンツ暗号化の最小単位として、セクタが挙げられるが、ヘッダ領域に鍵を保存するという目的の場合には、セクタの数だけ８バイト（ＤＥＳの場合）または１６バイト（トリプルＤＥＳ（Triple-DESの場合））の鍵情報が必要となるためヘッダのサイズが膨大になってしまい限られたメモリ領域のデータ領域を減少させてしまうことになり、実用上好ましくない。また、各セクタのデータ部分にそのセクタを暗号化するための鍵を格納する方法をとればヘッダサイズに影響を及ぼすことはないが、鍵の領域にはデータを置けなくなるためデータサイズが目減りしてしまうことと、万一、制御部側でファイルシステムを持つようなシステムの場合にはファイルシステム自体に大幅な変更を必要とする。

【０１６１】

そこで、本発明のシステムでは、先に説明した各コンテンツの属性情報であるセキュリティヘッダ（図７参照）の中に例えば、メディアの１ブロックあたりのセクタ数Ｍに対応するＭ個の鍵情報を格納し、これらを各セクタに対する暗号化鍵として適用する（図8参照）。図７に示したセキュリティヘッダ中のＫｃ__Encrypted０～Ｋｃ__Encrypted３１が３２個の暗号化鍵Ｋｃを示す。なお、［Encrypted］は、それぞれの鍵Ｋｃが暗号化されて格納されていることを示す。これらの複数の鍵の中からセクタのブロック内位置によって鍵を選択してセクタ対応の

暗号化鍵として用いる構成とした。

【０１６２】

図２７に、コンテンツのヘッダ情報としてコンテンツに対応して生成されるセキュリティヘッダにおける鍵格納構成と、各格納鍵と、各鍵の適用対象となるメモリ内の各セクタとの対応を説明する図を示す。図２７（ａ）が先に図７を用いて説明したセキュリティヘッダ内の鍵格納構成を簡略化して示した図である。図２７（ａ）のセキュリティヘッダには、Ｋｃ（０）～Ｋｃ（Ｍ－１）までのＭ個の鍵（コンテンツキー）が格納されている。ヘッダには鍵以外にもバージョン、コンテンツタイプ等の各種情報が格納され、さらにヘッダ情報の改竄チェック用のＩＣＶが格納されている。

【０１６３】

このＭ個のコンテンツキーは、例えば図２７（ｂ）に示すように各々が各セクタに対応付けられて各セクタに格納するデータの暗号化に使用される。先に図３を用いて説明したように、ブロック単位での消去を行なうフラッシュメモリは、図２７（ｂ）に示すようにデータ格納領域がブロック単位に分割され、各ブロックはさらに複数セクタに分割されている。例えば鍵Ｋｃ（０）を、メモリの各ブロックのセクタ０に格納するデータの暗号化鍵として適用し、鍵Ｋｃ（ｓ）を、メモリの各ブロックのセクタｓに格納するデータの暗号化鍵とする。さらに、鍵Ｋｃ（Ｍ－１）を、メモリの各ブロックのセクタＭ－１に格納するデータの暗号化鍵として適用する。

【０１６４】

このように、セクタに対応して異なる暗号鍵を適用してデータを格納することにより格納データ（ｅｘ．コンテンツ）のセキュリティが高められる。すなわち、コンテンツ全体を１つの鍵で暗号化した場合は、鍵漏洩によるコンテンツ全体の復号が可能となるのに対し、本構成によれば、１つの鍵の漏洩によってデータ全体を復号することは不可能であるからである。

【０１６５】

暗号化アルゴリズムは、例えば１つの暗号鍵によるＤＥＳ暗号化処理を実行するシングルＤＥＳが適用される。また、シングルＤＥＳではなく、暗号化に２つ

以上の鍵を使用するトリプルＤＥＳ（Triple ＤＥＳ）を適用した暗号化構成としてもよい。

【０１６６】

トリプルＤＥＳ（Triple ＤＥＳ）の詳細構成例を図２８に示す。図２８（ａ）、（ｂ）に示すようにトリプルＤＥＳ（Triple ＤＥＳ）としての構成には、代表的には以下のような２つの異なる態様がある。図２８（ａ）は、２つの暗号鍵を用いた例を示すものであり、鍵１による暗号化処理、鍵２による復号化処理、さらに鍵１による暗号化処理の順に処理を行う。鍵は、Ｋ１、Ｋ２、Ｋ１の順に２種類用いる。図２８（ｂ）は３つの暗号鍵を用いた例を示すものであり、鍵１による暗号化処理、鍵２による暗号化処理、さらに鍵３による暗号化処理の順に処理を行い３回とも暗号化処理を行う。鍵は、Ｋ１、Ｋ２、Ｋ３の順に３種類の鍵を用いる。このように複数の処理を連続させる構成とすることで、シングルＤＥＳに比較してセキュリティ強度を向上させることが可能である。

【０１６７】

図２９に、メモリに格納するデータの各セクタ毎に異なる２つの暗号鍵のペアを適用してトリプルＤＥＳによる暗号化処理を行なった構成例を示す。図２９に示すように、各ブロックのセクタ０は、鍵Ｋｃ（０）とＫｃ（１）の２つの鍵を用いてトリプルＤＥＳ暗号化を行ない、セクタｓは、鍵Ｋｃ（ｓ）とＫｃ（ｓ＋１）の２つの鍵を用いてトリプルＤＥＳ暗号化を行ない、セクタＭ－１は、鍵Ｋｃ（Ｍ－１）とＫｃ（０）の２つの鍵を用いてトリプルＤＥＳ暗号化を行なう。この場合でも、ヘッダに格納する鍵数は、Ｍ個であり、図２７（ａ）で示した鍵格納数を増加させる必要はなく、セキュリティを高めることが可能となる。

【０１６８】

さらに、図３０に異なる態様でのデータ暗号化構成例を示す。図３０は、メモリの各ブロックの２つの連続するセクタ領域を１つの暗号化ブロックとして、２つの鍵を用いてトリプルＤＥＳ暗号化を行なった態様である。図３０に示すように、各ブロックのセクタ０とセクタ１は、鍵Ｋｃ（０）とＫｃ（１）の２つの鍵を用いてトリプルＤＥＳ暗号化を行ない、セクタ２ｓとセクタ２ｓ＋１は、鍵Ｋｃ（２ｓ）とＫｃ（２ｓ＋１）の２つの鍵を用いてトリプルＤＥＳ暗号化を行な

い、セクタＭ－２とセクタＭ－１は、鍵Ｋｃ（Ｍ－２）とＫｃ（Ｍ－１）の２つの鍵を用いてトリプルＤＥＳ暗号化を行なう。このように複数のセクタに同一の暗号化処理を適用することで暗号化プロセスまたは復号プロセスの処理軽減を可能とすることができる。

【０１６９】

図２７，図２９，図３０に示す例の他にも、ヘッダに複数鍵を格納し、その複数鍵から選択した鍵を用いてセクタ毎の暗号化を実行する構成としては様々な構成が可能である。例えば、図２７，２９，３０では、セクタ数と同数の鍵をヘッダに格納する構成としているが、例えばセクタ数がＭのとき、格納鍵数をＮ（Ｎ＜Ｍ）として、セクタ０とセクタｓは同じ鍵で暗号化する等の構成としてもよい。また格納鍵数をＬ（Ｌ＞Ｍ）として、各セクタごとに全く異なる複数の鍵セットによるトリプルＤＥＳを適用する構成としてもよい。

【０１７０】

［セクタ単位の改竄チェック値（ＩＣＶ）の付加構成］

次に、セクタ単位の改竄チェック値（ＩＣＶ）の付加構成について説明する。複数セクタにまたがって構成されるデータについて、その正当性を確認する場合、一般には、コンテンツデータ全体の最後などに前述した改竄チェック値（ＩＣＶ）を付加させる構成とするのが一般的であった。このようなデータ全体のＩＣＶの付加構成においては、データを構成している各セクタ単位で、正当性を確認することができない。

【０１７１】

またＩＣＶを格納する場合、実データであるコンテンツの格納領域と同領域にＩＣＶを入れ込むと、その分データ部として使用できる領域が減ってしまう。もし、各セクタにセクタ内のデータに対してセクタ毎のＩＣＶを入れ込むと、デバイスのファイルシステムはデータ部単位でデータを読み出す処理を実行するために、実際に使用されるデータをＩＣＶから切り離して取り出すための処理、すなわち一度、読み出したデータ部内のセクタ内のＩＣＶを取り除く処理と、取り出したセクタ内のデータを複数セクタで連結する処理を実行することが必要となり、その処理を実行するためのファイルシステムを新たに構築することが必要とな

る。さらに、これらのＩＣＶチェックを制御部で行うとなると、制御部にその処理の分の負荷がかかってしまう。

【０１７２】

本発明のデータ処理装置においては、セクタ毎にデータ改竄チェックを可能とするため、セクタ毎にＩＣＶを設定し、そのＩＣＶ設定位置を実データ領域ではなく、デバイスのファイルシステムによって読み取られない領域として予め設定されている冗長部領域とした。冗長部にＩＣＶを置く構成とすることで、データ内にＩＣＶを置く必要がなくなり、データ部の領域が多く利用できる。また、冗長部にＩＣＶを置くことで、データ部とＩＣＶの切り分け・データ連結処理が不必要となるために、データ読み出しの連続性が保たれる。

【０１７３】

データを読み出す時には、メモリインタフェース部３００（図２参照）でセクタ毎のＩＣＶチェック処理を実行し、改竄ありと判定され無効なデータである場合は制御部２０５（図２参照）への転送を実行しない。また、データ書き込み時には、メモリインタフェース部３００において各セクタのＩＣＶを計算して、冗長部に書きこむ処理を実行する。

【０１７４】

なお、各セクタでＩＣＶを付加するかしないかを、セキュリティヘッダ（Security Header）に記述して指定する。この構成については、図７のセキュリティヘッダ構成の説明中に示したように、セキュリティヘッダ中のＩＣＶフラグ（ICV Flag）が、ブロック内のセクタ数（32セクタ）分のフラグを持ち、ブロック内の各セクタのＩＣＶ付加・非付加を示す。例えば０：ＩＣＶなし、１：ＩＣＶあり、として設定される。

【０１７５】

各セクタのデータ利用部と冗長部構成を図３１に示す。図３１（ａ）のように、メモリ（フラッシュメモリ）に格納されるデータは複数のセクタ領域を持つブロック単位領域に分割して格納される。（ｂ）に示すように、各セクタはデバイスのファイルシステムによって実データ（ｅｘ．コンテンツ）として読み取られる例えば５１２あるいは１０２４バイトのデータ利用部と、ファイルシステムに

よっては読み取られないＥＣＣ（Error Correction Code）等の情報を格納した冗長部とによって構成される。

【０１７６】

この冗長部の容量は例えば１６バイト、あるいは２０バイトの予め決められた領域であり、デバイスのファイルシステムは、この冗長部を非データ領域として認識し、データ（コンテンツ）読み取り処理においては読み取らない。一般に、冗長部に格納されるＥＣＣは、冗長部全体を使用せず、冗長部には非使用領域（リザーブ領域）が存在する。このリザーブ領域に各セクタの改竄チェック値（ＩＣＶ）を格納する。

【０１７７】

冗長部にＩＣＶを格納した場合のデバイスのファイルシステムによるデータ部の連結処理は、図３１（ｃ）に示すように、純粋にデータとして使用するものだけが格納されたデータ部の連結を行なうのみの従来のデータ連結処理と同様の処理が可能となる。従って、デバイスのファイルシステムは、冗長部を除くデータ部領域を単に連結すればよく、新たな処理は何ら必要としない。

【０１７８】

本構成により、複数のセクタで構成されるデータのセクタ単位でデータの正当性の検証することが出来る。また、改竄チェック用のＩＣＶを冗長部に入れることで、データ用に使えるデータ領域をそのまま活用することが出来る。また、制御部には、ＩＣＶチェックの結果、正しい（改竄なし）と判定された正しいセクタのみが送信される。また、ＩＣＶチェックがメモリインタフェース部にて行われるので、制御部の負担がかからない等の効果がある。

【０１７９】

[メディア内の個別鍵によるコンテンツ鍵の保存処理]

次に、メディア内の個別鍵によるコンテンツ鍵の保存処理構成について説明する。先に、図７を用いて説明したように、コンテンツに対応して構成されるセキュリティヘッダには、セクタ対応の暗号鍵としての複数のコンテンツキー（Ｋｃ__Encryptedxx）、およびコンテンツチェック値生成鍵（Ｋｉｃｖ__Encrypted）が暗号化されて格納されている。

【０１８０】

これらの鍵の暗号化の１つの態様は、予めデバイスのメモリインタフェースのメモリ部３２１（図４参照）に格納されている配送鍵Ｋｄｉｓｔによって暗号化して格納する構成がある。例えば、Ｋｃ__Encrypted０＝Ｅｎｃ（Ｋｄｉｓｔ，Ｋｃ（０））である。ここで、Ｅｎｃ（ａ，ｂ）は、ｂをａで暗号化したデータであることを示す。このように、それぞれの鍵をデバイスの配送鍵Ｋｄｉｓｔを用いて暗号化してセキュリティヘッダに格納する構成が１つの構成である。

【０１８１】

さらに、メディア２、すなわち暗号処理部を持ち、デバイスとの相互認証を実行してコンテンツ処理を実行するメディアにおいて、メディア２の固有鍵を用いてメディア２に格納するコンテンツに関するコンテンツキー、ＩＣＶ生成鍵を暗号化する態様がある。以下、メディア２の固有鍵、ここではメデイア２保存鍵Ｋｓｔｏを用いて暗号化したコンテンツキー、コンテンツＩＣＶ生成鍵をセキュリティヘッダに格納する処理について説明する。

【０１８２】

メディア２保存鍵Ｋｓｔｏは、図２に示したようにメディア２，２３０のメディア２コントローラ２３１の内部メモリ２３５に格納されている。従って、メディア２保存鍵Ｋｓｔｏを使用したコンテンツキー、ＩＣＶ生成鍵の暗号化処理、復号処理はメディア２側で実行される。メディア２を装着したデバイスが、メディア２のコンテンツ利用に際し、コンテンツキー、ＩＣＶ生成鍵を取得、あるいはセキュリティヘッダへの格納処理を実行する場合は、メデイア２側で鍵の暗号化、復号処理を実行することが必要となる。本発明のデータ処理装置においては、これらをＣＢＣ（Cipher Block Chaining）モードで処理することを可能とした。

【０１８３】

図３２にＣＢＣモードにおける鍵の暗号化処理構成を示す。この暗号化処理は、メディア２の暗号処理部２３６（図２参照）において実行される。内部メモリ２３５に格納された初期値ＩＶ__keysと、コンテンツチェック値生成鍵Ｋｉｃｖ__contとの排他論理和を実行し、その結果をメディア２の内部メモリ２３５に格

納された保存鍵Ｋｓｔｏを適用したＤＥＳ暗号化を行ない、その結果をＫｉｃｖ_cont Encryptedとしてヘッダに格納する。さらに、Ｋｉｃｖ_cont Encryptedと、セクタ（０）に対応するセクタ対応コンテンツキーＫｃ（０）との排他論理和を実行し、その結果をメディア２の内部メモリ２３５に格納された保存鍵Ｋｓｔｏを適用したＤＥＳ暗号化を行ない、その結果をＫｃ（０）Encryptedとしてヘッダに格納する１つの暗号化コンテンツキーとする。さらに、Ｋｃ（０）Encryptedと、セクタ（１）に対応するセクタ対応コンテンツキーＫｃ（１）との排他論理和を実行し、その結果に対して保存鍵Ｋｓｔｏを適用したＤＥＳ暗号化を行ない、その結果をＫｃ（１）Encryptedとする。以下、これらの処理を繰り返し実行して、ヘッダ格納用の鍵データとする。

【０１８４】

次に、図３３にＣＢＣモードにおける鍵の復号処理構成を示す。この復号処理は、メディア２の暗号処理部２３６（図２参照）において実行される。まず、Ｋｃ（０）Encryptedに対して、メディア２の内部メモリ２３５に格納された保存鍵Ｋｓｔｏを適用したＤＥＳ復号処理を行ない、その結果を内部メモリ２３５に格納された初期値ＩＶ__keysと排他論理和することにより、セクタ（０）に対応するセクタ対応コンテンツキーＫｃ（０）が出力される。さらに、Ｋｃ（１）Encryptedに対して、保存鍵Ｋｓｔｏを適用したＤＥＳ復号処理を行ない、その結果をコンテンツキーＫｃ（０）Encryptedと排他論理和することにより、セクタ（１）に対応するセクタ対応コンテンツキーＫｃ（１）が出力される。以下、これらの処理を繰り返し実行して、コンテンツキーを取得する。なお、図には、コンテンツキーのみを出力データとした例を示しているが、コンテンツ改竄チェック値生成鍵（Ｋｉｃｖ__Encrypted）についても同様の処理が適用可能であり、暗号化されたコンテンツ改竄チェック値生成鍵（Ｋｉｃｖ__Encrypted）からコンテンツ改竄チェック値生成鍵（Ｋｉｃｖ）の生成が可能である。

【０１８５】

上述のセクタ対応コンテンツキーＫｃ（ｘｘ）またはコンテンツ改竄チェック値生成鍵（Ｋｉｃｖ）の暗号化、復号処理は、多くの場合、メディア２を装着したデバイスからのコマンドに基づいて実行される。この場合、デバイスとメディ

ア２間では前述した相互認証が実行され、相互認証処理が成立したことを条件としてコンテンツ再生、格納等の様々な処理が実行され、その一連のコンテンツ処理の１つとして上述のコンテンツキーの復号、暗号化処理が実行されることになる。復号された鍵（ｅｘ．コンテンツキーＫｃ（ｘｘ））をデバイスとメディア２間において転送する場合は、相互認証時に生成したセッションキーＫｓｅｓで暗号化される。このセッションキーＫｓｅｓによる暗号化、復号処理もＣＢＣモードを適用することで、よりセキュリテイを高めることが可能となる。

【０１８６】

図３４にメディア２において、セキュリティヘッダに格納された鍵をＤＥＳ－ＣＢＣモードで復号し、復号した鍵データをさらにセッションキーＫｓｅｓを適用してＤＥＳ－ＣＢＣモードで暗号化する処理構成を示す。図３４の上段は、図３３と同様の構成であり、セキュリティヘッダから取り出した暗号化されたコンテンツキーを順次ＤＥＳ復号部に入力してメディア２の保存鍵Ｋｓｔｏを適用して復号処理を実行し、出力結果を初期値、または入力データ列の前データと排他論理和して、出力結果としてのコンテンツキーを取得する。

【０１８７】

これらの出力された結果をさらに、デバイスとの相互認証時に生成したセッションキーＫｓｅｓを適用したＤＥＳ－ＣＢＣモードでの暗号化処理を実行する。その結果得られたＳＥ０～ＳＥＭ－１：Ｋｃ（０）Encrypted～Ｋｃ（Ｍ－１）Encryptedをデバイスに送信する。デバイス側では、受信したデータ列Ｋｃ（０）Encrypted～Ｋｃ（Ｍ－１）Encryptedについて、メディア２との相互認証時に生成したセッションキーＫｓｅｓを適用して、図３３と同様のＤＥＳ－ＣＢＣモードでの復号処理を実行することによりコンテンツキーＫ（ｃ）を取得することができる。なお、図には、コンテンツキーのみを処理データとした例を示しているが、コンテンツ改竄チェック値生成鍵（Ｋｉｃｖ__Encrypted）についても同様に処理データとして構成することが可能である。

【０１８８】

[暗号化データの読み出し処理]

図３５以下のフローを用いて、暗号化されたデータのメディアからの読み出し

処理の詳細を説明する。なお、データの暗号化態様は、上述したようにセクタ毎に異なる鍵で暗号化した態様と、コンテンツ全体を１つの暗号化鍵で暗号化した態様とがあり、これらは、ヘッダの情報に基づいて判定される。図３５のフローにおいて左側はデバイスの制御部、右側はデバイスのメモリインタフェースの処理である。

【０１８９】

まず制御部は、読み出し対象となるコンテンツのヘッダファイルを読み出す（Ｓ７０１）。この処理は、前述の図２５のファイル読み出し処理フローに従った処理として実行される。次にヘッダセットコマンドと、読み出したヘッダファイルをメモリインタフェースに送信（Ｓ７０２）する。

【０１９０】

メモリインタフェースはヘッダセットコマンドを受信（Ｓ７０３）すると、ビジーフラグを１（ビジー）にセット（Ｓ７０４）し、ヘッダの改竄チェック値（ＩＣＶ）を検証（Ｓ７０５）する。ヘッダのＩＣＶチェックは、先に図１４を用いて説明したＩＣＶ生成処理において、セキュリティヘッダ検証値生成鍵Ｋｉｃｖ＿ｓｈと、初期値ＩＶｓｈを適用してヘッダの構成データを入力してＩＣＶ’を生成し、生成したＩＣＶ’と予めヘッダに格納されたＩＣＶとを照合する処理によって実行する。

【０１９１】

検証によりヘッダ改竄なしと判定（Ｓ７０６）されると、ヘッダ内の有効リボケーションリスト・バージョンが０でないかがチェック（Ｓ７０７）される。例えば、自デバイスで生成し格納したコンテンツをメモリに格納するとき等は、リボケーションリスト・バージョンを０として、再生処理等の際にリボケーションリストを非参照とした処理を実行可能とする。

【０１９２】

リボケーションリスト・バージョンが０の場合は、リボケーションリストを参照する必要がないのでステップＳ７１０に進む。バージョンが非０であるときは、現在セットされているリボケーションリストが、ヘッダのバージョンより古くないかをチェック（Ｓ７０８）し、古い場合は、Ｓ７１３に進み、ヘッダセット

成功フラグを０（ＮＧ）に設定して処理を終了する。セットされているリボケーションリストがヘッダのバージョンより古くなければ、ステップＳ７０９に進み、リボケーションリストを参照して、読み出し対象のコンテンツＩＤがないかを判定する。あった場合は読み出しを禁止する処理として、ステップＳ７１３でヘッダセット成功フラグを０（ＮＧ）として処理を終了する。

【０１９３】

リボケーションリストに読み出し対象コンテンツＩＤが記録されていなければ、ステップＳ７１０進み、ヘッダ情報に基づいて暗号化されたコンテンツキーＫｃと、コンテンツチェック値生成鍵Ｋｉｃｖ__contを復号する。なお、リボケーションリストは、先の図１６の起動時フローにおいて説明したように、起動時にメモリインタフェースにセットアップされ、セットアップ後は、メデイアの装着時、コンテンツの再生時においてメモリインタフェースにおいて継続的に利用可能としたリボケーションリストである。

【０１９４】

先に、図７他を用いて説明したようにセキュリティヘッダの中には、前述のセクタ毎に適用する暗号鍵としての複数のコンテンツキーＫｃ（０）～Ｋｃ（Ｍ－１）が暗号化されて格納されている。また、コンテンツの改竄チェック値（ＩＣＶ）を生成するためのコンテンツチェック値生成鍵Ｋｉｃｖ__contも暗号化されて格納されている。

【０１９５】

コンテンツの復号に先立ち、これらのコンテンツチェック値生成鍵Ｋｉｃｖ__contを復号してコンテンツの改竄チェックを実行する処理が必要であり、また、コンテンツキーＫｃ（０）～Ｋｃ（Ｍ－１）を復号する処理が必要となる。

【０１９６】

図３７に暗号化されたコンテンツキーＫｃ、コンテンツチェック値生成鍵Ｋｉｃｖ__contの復号処理フローを示す。図３７の各ステップについて説明する。図３７の処理は、デバイスのメモリインタフェースにおける処理である。図４の暗号処理部３２０において実行される。

【０１９７】

まず、暗号化コンテンツチェック値生成鍵Ｋｉｃｖ__contを復号対象として選定（Ｓ８０１）し、次に、ヘッダの暗号化フォーマットタイプ・フィールドの設定が０か否かを判定（Ｓ８０２）する。暗号化フォーマットが０である場合は、コンテンツ全体をセクタに係わらず１つの暗号化態様としたデータ構成であり、暗号化フォーマットタイプ・フィールドの設定が１である場合は、前述の図２７他で説明したセクタ単位の暗号化鍵を用いた方法である。セクタ単位の暗号化鍵を用いた方法である場合は、ステップＳ８０３に進み、セクタ毎に設定された暗号化コンテンツキー（Ｋｃ_Encrypted0～31）を復号対象にする。

【０１９８】

ステップＳ８０２で暗号化フォーマットが０であると判定された場合は、ステップＳ８０４でさらに、ヘッダの暗号化アルゴリズムフィールドをチェックして１（トリプルＤＥＳ）が０（シングルＤＥＳ）であるかを判定する。シングルＤＥＳである場合は、ステップＳ８０５で１つの暗号化コンテンツキー（Ｋｃ_Encrypted0）のみを復号対象として加え、トリプルＤＥＳである場合は、ステップＳ８０６で複数の暗号化コンテンツキー（Ｋｃ_Encrypted0，１）を復号対象として加える。

【０１９９】

次に、ステップＳ８０７において、ヘッダのコンテンツタイプフィールドの設定をチェックし、設定が２または３（メディア２の格納コンテンツ）でない場合は、ステップＳ８０８で、メモリ部３２１（図４参照）に格納された配送鍵Ｋｄｉｓｔで復号対象データ、すなわち、暗号化コンテンツチェック値生成鍵Ｋｉｃｖ__contと、１以上のコンテンツキーを復号する。

【０２００】

設定が２または３（メディア２の格納コンテンツ）である場合は、ステップＳ８０９で復号対象データ、すなわち、暗号化コンテンツチェック値生成鍵Ｋｉｃｖ__contと、１以上のコンテンツキーをメデイア２の保存鍵Ｋｓｔｏ（ＣＢＣモード）で復号する。この復号処理の詳細は、図３２、図３３、図３４を用いて説明した通りである。

【０２０１】

ステップＳ８０９におけるメディア２の保存鍵による暗号化コンテンツチェック値生成鍵Ｋｉｃｖ＿contと、１以上のコンテンツキーＫｃの復号処理について図３８のフローを用いて説明する。図３８のフローは、左側にデバイスのメモリインタフェース、右側にメディア２のコントローラ（図２参照）の処理を示している。

【０２０２】

まず、メモリインタフェースは、復号対象データＫ（０）～Ｋ（ｎ－１）（暗号化コンテンツチェック値生成鍵Ｋｉｃｖ＿contと、１以上のコンテンツキー）を設定（Ｓ１００１）し、ＣＢＣ復号初期化コマンドをメディア２コントローラに送信（Ｓ１００３）し、メディア２コントローラはＩＶＫｅｙｓをレジスタにセット（Ｓ１００５）する。その後、メモリインタフェースは、各鍵を順次送信（Ｓ１００４）し、メディア２コントローラが復号対象データＫ（ｉ）を受信（Ｓ１００５）する。

【０２０３】

次にメディア２コントローラは、受信した復号対象データＫ（ｉ）に対して、メディア２の保存鍵Ｋｓｔｏを用いたＣＢＣモードによる復号処理を実行（Ｓ１００７）し、復号された鍵データ（ｅｘ．複数のセクタ対応コンテンツキー）を取得（Ｓ１００８）する。次に、メディア２コントローラは、復号鍵データ列を、デバイスとの相互認証時に生成したセッションキーによってＣＢＣモードでの暗号化処理を実行し、データ列Ｋ’（ｉ）を生成して、結果をデバイスに送信（Ｓ１００９）する。ステップＳ１００７～Ｓ１００９の処理は、先に説明した図３４のＤＥＳ－ＣＢＣモードによる処理に基づいて実行される。

【０２０４】

デバイスのメモリインタフェースは、順次Ｋ’（ｉ）を受信し、すべてのデータを受信したことを確認の後、ＣＢＣ終了コマンドをメディア２コントローラに送信する。メディア２コントローラはＣＢＣ終了コマンドの受信によりレジスタをクリア（Ｓ１０１４）する。

【０２０５】

デバイスのメモリインタフェースは、メモリ部３２１（図４参照）に格納した

初期値ＩＶ_keysを用い、メディア２との相互認証時に生成したセッションキーＫｓｅｓを適用してＣＢＣモードでメディア２から受信したK'（ｉ）を復号（Ｓ１０１０～Ｓ１０１３，Ｓ１０１５）する。この復号処理は、先に説明した図３３の構成と同様の処理である。

【０２０６】

上記処理により、デバイスは、ヘッダに格納された暗号化されたコンテンツキーＫｃ、コンテンツチェック値生成鍵Ｋｉｃｖ__contを復号し、それぞれの鍵を取得することができる。

【０２０７】

次に図３５に戻り、暗号化ファイルの読み出し処理の続きを説明する。上記の鍵復号処理ステップであるステップＳ７１０を終了すると、ステップＳ７１１に進む。ステップＳ７１１では、デバイスのメモリインタフェースはヘッダを「読み出しヘッダ」として内部に設定し、ヘッダセット成功フラグを１（成功）にセットし、ビジーフラグを０（待機）（Ｓ７１４）設定する。コンテンツ読み出しに際しては、設定されたヘッダの情報に基づく処理が実行される。

【０２０８】

一方、制御部側は、ステップＳ７１５でステータス読み出しコマンドをメモリインタフェースに送信し、ビジーフラグが０（待機）（Ｓ７１６）であり、ヘッダセット成功フラグが１（成功）（Ｓ７１７）となったことを条件として次の処理（図３６）に進む。

【０２０９】

図３６のステップＳ７２１において、制御部は、ファイル割り当てテーブルから読み出し対象のコンテンツファイルのセクタアドレス（Ｓ（１）～Ｓ（ｋ））を取得し、メモリインタフェースに対して順次、セクタＳ（ｉ）読み出しコマンドを送信する。

【０２１０】

メモリインタフェースは、セクタＳ（ｉ）読み出しコマンドを受信（Ｓ７２４）すると、ビジーフラグを１（ビジー）に設定（Ｓ７２５）し、ヘッダ成功フラグが１（成功）であることを条件（Ｓ７２６）として次ステップに移行する。へ

ッダ成功フラグが１（成功）でない場合は、ステップＳ７３８に進み、読み出し成功フラグを０（ＮＧ）として処理を終了する。

【０２１１】

ヘッダ成功フラグが１（成功）である場合は、受信セクタＳ（ｉ）が内部メモリか、外部メモリであるかを判定（Ｓ７２７）し、外部メモリである場合は、メディア１かメディア２のセットフラグが１（メディアが有効にセットされていることを示す）であるかを判定（Ｓ７２８）し、セットフラグが１である場合には、さらにブロックパーミッション・テーブル（ＢＰＴ）を参照して、ＢＰＴが読み出し対象であるセクタＳ（ｉ）を読み出し許可対象ブロックとして設定しているかを判定（Ｓ７２９）する。ＢＰＴに読み出し許可ブロックの設定がある場合には、外部メモリから該当セクタのデータを読み出す（Ｓ７３０）。

【０２１２】

なお、読み出し対象データがＢＰＴによる管理のなされていない内部メモリ内のデータである場合は、ステップＳ７２８，Ｓ７２９はスキップする。ステップＳ７２８、Ｓ７２９の判定がＮｏである場合、すなわちセクタＳ（ｉ）を格納したメディアのセットフラグが１でない場合、または、ＢＰＴにセクタＳ（ｉ）の読み出し許可が設定されていない場合には、ステップＳ７３８に進み、読み出しエラーとして読み出し成功フラグが０にセットされる。

【０２１３】

ステップＳ７２６～Ｓ７２９の判定ブロックにおいて、対象セクタＳ（ｉ）の読み出しが実行可と判定されると、メモリから該当セクタが読み出され、セクタに対応して設定されている冗長部の誤り訂正符号に基づく誤り訂正処理が実行（Ｓ７３１）され、誤り訂正が成功した（Ｓ７３２）ことを確認する。次に、ヘッダのＩＣＶフラグ（図7参照）を参照し、読み出し対象セクタが改竄チェック値（ＩＣＶ）による処理対象であるかを判定する。先に図３１を用いて説明したように各セクタは、その冗長部に改竄チェック用のＩＣＶを格納しており、セクタ単位での改竄チェックが可能である。

【０２１４】

ＩＣＶによる改竄チェックの対象である場合は、ステップＳ７３４において、

ステップＳ７１０の復号処理によって得たコンテンツチェック値生成鍵Ｋｉｃｖ__contと、初期値ＩＶｃｏｎｔを適用し改竄チェック対象データ（セクタデータ）を入力して図１４を用いて説明したＩＣＶ生成処理を実行し、ＩＣＶ’を求め、セクタの冗長部に格納されているＩＣＶとの照合を行ない一致していれば改竄なしと判定する。

【０２１５】

ＩＣＶチェックにより改竄なしと判定されると、ステップＳ７３７に進み、ヘッダ情報に基づいてデータの復号処理を実行して読み出し成功フラグを１（成功）に設定し、復号データをバッファに格納する。

【０２１６】

また、制御部は、ステップＳ７４０～Ｓ７４６において、メモリインタフェースのステータスを読み出して、ビジーフラグが０の状態において、読み出し成功フラグが１であることを条件として読み出しデータをバッファから取り出して保存し、アドレスを順次インクリメントして、データを順次バッファから取り出して保存する処理を繰り返し実行し、すべての読み出し対象セクタを保存した後、全読み出しセクタデータからファイルを構成して処理を終了する。

【０２１７】

図３６のステップＳ７３６のデータ部復号処理の詳細を図３９を用いて説明する。この復号処理はデバイスのメモリインタフェースの暗号処理部３２０（図４参照）において実行される。

【０２１８】

まず、復号対象のデータ格納セクタ位置をｓ（０≦ｓ≦３１（セクタ数３２の場合））とする（Ｓ１１０１）。次にそのセクタが暗号化対象であるかをチェック（Ｓ１１０２）する。このチェックは、セキュリティヘッダ（図７参照）の暗号化フラグ（Encryption Flag）に基づいて判定される。暗号化対象でない場合は、復号処理は実行されず、処理は終了する。暗号化対象である場合は、暗号化フォーマットタイプをチェック（Ｓ１１０３）する。これはセキュリティヘッダ内の暗号化フォーマットタイプ（Encryption Format Type）の設定をチェックするものであり、図８で説明したコンテンツ全体を１つの暗号化態様としているか

、各セクタに異なる鍵を用いた暗号化処理を行なっているかを判定する。

【０２１９】

暗号化フォーマットタイプ（Encryption Format Type）の設定値が０の場合は、コンテンツ全体を１つの暗号化態様としている場合である。この場合は、ステップＳ１１０４において、暗号化アルゴリズム（Encryption Algorithm）の判定を行なう。暗号化アルゴリズムは、シングルＤＥＳかトリプルＤＥＳ（図２８参照）かを設定しているものであり、シングルＤＥＳであると判定された場合は、１つのコンテンツキーＫｃ（０）を適用して暗号化コンテンツの復号処理を実行（Ｓ１１０６）する。トリプルＤＥＳであると判定された場合は、２つのコンテンツキーＫｃ（０）、Ｋｃ（１）を適用して暗号化コンテンツの復号処理を実行（Ｓ１１０７）する。

【０２２０】

一方、ステップＳ１１０３で、暗号フォーマットタイプ（Encryption Format Type）の設定値が１の場合は、各セクタに異なる鍵を用いた暗号化処理を行なっている場合である。この場合は、ステップＳ１１０５において、暗号化アルゴリズム（Encryption Algorithm）の判定を行なう。暗号化アルゴリズムは、シングルＤＥＳかトリプルＤＥＳ（図２８参照）かを設定しているものであり、シングルＤＥＳであると判定された場合は、各セクタ（ｓ）に対応して設定されたコンテンツキーＫｃ（ｓ）を各セクタに適用して暗号化コンテンツの復号処理を実行（Ｓ１１０８）する。トリプルＤＥＳであると判定された場合は、２つのコンテンツキーＫｃ（ｓ）、Ｋｃ（ｓ＋１ｍｏｄ３２）を適用して各セクタ毎の暗号化コンテンツの復号処理を実行（Ｓ１１０９）する。

【０２２１】

セクタデータの復号処理の異なる処理態様を図４０に示す。図４０において、ステップＳ１２０１～Ｓ１２０８は、図３９の各ステップＳ１１０１～Ｓ１１０８と同様である。ステップＳ１２０９～Ｓ１２１１が図３９とは異なる。

【０２２２】

ステップＳ１２０５において、暗号化アルゴリズムがトリプルＤＥＳであると判定された場合、ステップＳ１２０９においてセクタＮｏ．（ｓ）を判定し、ｓ

が奇数である場合は、ｓ＝ｓ－１の更新を実行（Ｓ１２１０）し、各セクタに適用する鍵をＫｃ（ｓ），Ｋｃ（ｓ＋１）としてトリプルＤＥＳによる復号処理（Ｓ１２１１）を実行する。

【０２２３】

以上、暗号化されて格納されたデータの復号処理を伴う再生処理は、図３５～図４０を用いて説明したようなプロセスにより実行される。

【０２２４】

[データの暗号化書き込み処理]

次に、図４１以下のフローを用いて、メディアに対するデータの暗号化書き込み処理プロセスの詳細を説明する。なお、データの暗号化態様は、上述したようにセクタ毎に異なる鍵で暗号化した態様と、コンテンツ全体を１つの暗号化鍵で暗号化した態様とがある。これらは、ヘッダ情報に設定される。図４１のフローにおいて左側はデバイスの制御部、右側はデバイスのメモリインタフェースの処理である。

【０２２５】

まず制御部は、読み出し対象となる格納コンテンツに対応するヘッダ生成コマンドとヘッダ情報としてのパラメータをメモリインタフェースに送信する。（Ｓ１３０１）。

【０２２６】

メモリインタフェースはヘッダ生成コマンドを受信（Ｓ１３０２）すると、ビジーフラグを１（ビジー）にセット（Ｓ１３０３）し、受信パラメータが許容値内であるかを判定（Ｓ１３０４）する。メモリインタフェースは、予めヘッダに設定可能なパラメータ範囲を有しており、受信パラメータと比較し、受信パラメータが設定可能範囲を超えている場合は、ステップＳ１３１０においてヘッダ生成成功フラグを０（ＮＧ）に設定して処理を終了する。受信パラメータが許容値内である場合は、ヘッダの有効リボケーションリストバージョンを０に設定（Ｓ１３０５）し、リボケーションリストの非参照でのデータ処理を可能とする。有効リボケーションリストバージョンを０として設定するのは、自デバイスでの格納処理を行なったコンテンツについては正当なコンテンツであることが保証され

ているとの前提により、リボケーションリストの非参照でのデータ処理（再生）を可能とする設定を行なうものである。

【０２２７】

なお、書き込みコンテンツが例えば通信手段を介して外部から受信したコンテンツであり、受信コンテンツに識別子が付加され参照すべきリボケーションリストバージョンをヘッダに格納しておりデバイス内部のリボケーションリストとの照合が可能であれば、上記処理の代わりに、先に図３５を用いて説明したファイル復号読み出し処理において実行されるステップＳ７０７～Ｓ７０９と同様のリボケーションリストを用いた識別子照合処理を行なってもよい。

【０２２８】

次に、ステップＳ１３０６において、ヘッダ情報に基づいてコンテンツキーＫｃ、コンテンツ改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ＿contを生成、暗号化する。ステップＳ１３０６のコンテンツキーＫｃ、コンテンツ改竄チェック値生成鍵Ｋｉｃｖ＿contを生成、暗号化処理の詳細を図４３に示す。図４３の処理は、デバイスのメモリインタフェースの暗号処理部３２０（図４参照）において実行される。図４３のフローについて説明する。

【０２２９】

まず、暗号化コンテンツチェック値生成鍵Ｋｉｃｖ＿contを、例えば乱数に基づいて生成し、暗号化対象とし（Ｓ１４０１）、次に、ヘッダの暗号化フォーマットタイプ・フィールドの設定が０か否かを判定（Ｓ１４０２）する。暗号化フォーマットが０である場合は、コンテンツ全体をセクタに係わらず１つの暗号化態様とする構成であり、暗号化フォーマットタイプ・フィールドの設定が１である場合は、前述の図２７他で説明したセクタ単位の暗号化鍵を用いる方法である。セクタ単位の暗号化鍵を用いる場合は、ステップＳ１４０３に進み、セクタ毎に設定されたコンテンツキー（Ｋｃ（０）～Ｋｃ（31）（セクタ数３２の場合））を生成して暗号化対象とする。

【０２３０】

ステップＳ１４０４で暗号化フォーマットが０であると判定された場合は、ステップＳ１４０４でさらに、ヘッダの暗号化アルゴリズムフィールドをチェック

して１（トリプルＤＥＳ）か０（シングルＤＥＳ）であるかを判定する。シングルＤＥＳである場合は、ステップＳ１４０５で１つのコンテンツキー（Ｋｃ（０））を生成して暗号化対象として加え、トリプルＤＥＳである場合は、ステップＳ１４０６で複数のコンテンツキー（Ｋｃ（０）、Ｋｃ（１））を生成して暗号化対象として加える。

【０２３１】

次に、ステップＳ１４０７において、ヘッダのコンテンツタイプフィールドの設定をチェックし、設定が２または３（メディア２の格納コンテンツ）でない場合は、ステップＳ１４０８で、メモリ部３２１（図４参照）に格納された配送鍵Ｋｄｉｓｔでデータ、すなわち、コンテンツチェック値生成鍵Ｋｉｃｖ__contと、１以上のコンテンツキーを暗号化する。

【０２３２】

設定が２または３（メディア２の格納コンテンツ）である場合は、ステップＳ１４０９でデータ、すなわち、コンテンツチェック値生成鍵Ｋｉｃｖ__contと、１以上のコンテンツキーをメデイア２の保存鍵Ｋｓｔｏ（ＣＢＣモード）で暗号化する。この暗号化処理の詳細は、図３２、図３３、図３４を用いて説明した通りである。

【０２３３】

ステップＳ１４０９におけるメディア２の保存鍵によるコンテンツチェック値生成鍵Ｋｉｃｖ__contと、１以上のコンテンツキーＫｃの暗号化処理について図４４のフローを用いて説明する。図４４のフローは、左側にデバイスのメモリインタフェース、右側にメディア２のコントローラ（図２参照）の処理を示している。

【０２３４】

まず、デバイス側のメモリインタフェースは、暗号化対象データＫ（０）～Ｋ（ｎ－１）（コンテンツチェック値生成鍵Ｋｉｃｖ__contと、１以上のコンテンツキー）を設定（Ｓ１５０１）し、メディア２との相互認証時に生成したセッションキーを適用し、メモリ部３２１に格納した初期値ＩＶ_keysを用いてＤＥＳ－ＣＢＣモードによる暗号化対象データＫ（０）～Ｋ（ｎ－１）の暗号化を実行

し、データK’（０）～K’（ｎ－１）を生成（Ｓ１５０２）する。この暗号化処理は、先に説明した図３２と同様の処理構成において実行される。次に、メモリインタフェースは、ＣＢＣ暗号化初期化コマンドをメディア２コントローラに送信する。メディア２は、メディア２の内部に格納している初期値ＩＶ＿ｋｅｙｓをレジスタにセット（Ｓ１５０６）する。その後、メモリインタフェースは、各鍵を順次送信（Ｓ１５０５）する。

【０２３５】

メディア２コントローラは、データK’（ｉ）を受信（Ｓ１５０７）し、受信したデータK’（ｉ）に対して、デバイスとの相互認証時に生成したセッションキーによってＣＢＣモードでの復号処理を実行（Ｓ１５０８）し、復号された鍵データ（ｅｘ．複数のセクタ対応コンテンツキー）を取得（Ｓ１５０９）する。次に、メディア２コントローラは、復号鍵データ列を、メディア２の保存鍵Ｋｓｔｏを用いたＣＢＣモードによる暗号化処理を実行し、データ列K”（ｉ）を生成して、結果をデバイスに送信（Ｓ１５１０）する。ステップＳ１５０７～Ｓ１５１０の処理は、先に説明した図３４のＤＥＳ－ＣＢＣモードによる処理に基づいて実行される。

【０２３６】

デバイスのメモリインタフェースは、順次K”（ｉ）を受信し、すべてのデータを受信したことを確認の後、ＣＢＣ終了コマンドをメディア２コントローラに送信（Ｓ１５１１～Ｓ１５１４）する。メディア２コントローラはＣＢＣ終了コマンドの受信によりレジスタをクリア（Ｓ１５１５）する。

【０２３７】

デバイスのメモリインタフェースは、メディア２から受信したK”（０）～K”（ｎ－１）をヘッダ格納用の暗号化鍵データとする。上記処理により、デバイスは、ヘッダに格納する暗号化されたコンテンツキーＫｃ、コンテンツチェック値生成鍵Ｋｉｃｖ＿contを取得することができる。

【０２３８】

図４１に戻り、ファイルの暗号化書き込み処理の説明を続ける。ステップＳ１３０６において、上述のヘッダ格納鍵の生成、暗号化が終了すると、メモリイン

タフェースは生成したヘッダデータに基づく改竄チェック値ＩＣＶを生成（Ｓ１３０７）する。セキュリティヘッダのチェック値であるＩＣＶ＿ｓｈは、メモリ部３２１（図４参照）に格納された初期値ＩＶｓｈと、セキュリテイヘッダ改竄チェック値生成鍵Ｋｉｃｖ＿ｓｈを用いて、先に図１４を用いて説明したＩＣＶ生成構成に基づいて生成される。次に、ステップＳ１３０８で、生成されたヘッダを書き込みヘッダとして内部に保存し、ステップＳ１３０９でヘッダ生成成功フラグを１（成功）としてビジーフラグを０（待機）として処理を終了する。

【０２３９】

一方、制御部側は、ステップＳ１３１２でステータス読み出しコマンドをメモリインタフェースに送信し、ビジーフラグが０（待機）（Ｓ１３１３）であり、ヘッダ生成成功フラグが１（成功）（Ｓ１３１４）となったことを条件として、バッファからヘッダを読み出し、通常のファイルとしてメディアに保存（Ｓ１３１５）後、次の処理（図４２）に進む。

【０２４０】

図４２のステップＳ１３２１において、制御部は、書き込み対象のコンテンツファイルをセクタに分割する。分割されたデータをＤ（１）～Ｄ（ｋ）とする。制御部は、次に各データＤ（ｉ）の書き込みセクタＳ（ｉ）を設定して、メモリインタフェースにセクタＳ（ｉ）の暗号化書き込みコマンドと、データＤ（ｉ）を順次送信（Ｓ１３２１～Ｓ１３２４）する。メモリインタフェースは、セクタＳ（ｉ）暗号化書き込みコマンドを受信（Ｓ１３２５）すると、ビジーフラグを１（ビジー）に設定（Ｓ１３２６）し、ヘッダ生成成功フラグが１（成功）である（Ｓ１３２７）ことを条件として次ステップに進む。

【０２４１】

次に、メモリインタフェースは、受信セクタＳ（ｉ）が内部メモリか、外部メモリであるかを判定（Ｓ１３２８）し、外部メモリである場合は、メディア１かメディア２のセットフラグが１（メディアが有効にセットされていることを示す）であるかを判定（Ｓ１３２９）し、セットフラグが１である場合には、さらにブロックパーミッション・テーブル（ＢＰＴ）を参照して、ＢＰＴが書き込み対象であるセクタＳ（ｉ）を書き込み許可対象ブロックとして設定しているかを判

定（Ｓ１３３０）する。ＢＰＴに書き込み許可ブロックの設定がある場合には、セクタに対応して設定する誤り訂正符号を生成（Ｓ１３３１）する。

【０２４２】

次に、その書き込みセクタがＩＣＶ設定セクタであるか否かをヘッダ情報（ＩＣＶフラグ）に基づいて判定（Ｓ１３３２）し、ＩＣＶ対象である場合は、コンテンツＩＣＶ生成鍵Ｋｉｃｖ＿contに基づいてセクタデータに対するＩＣＶを生成（Ｓ１３３３）する。

【０２４３】

次に、メモリインタフェースは、ヘッダ情報に基づくデータの暗号化を実行（Ｓ１３３４）する。ステップＳ１３３４のデータ部暗号化処理の詳細を図４５を用いて説明する。この暗号化処理はデバイスのメモリインタフェースの暗号処理部３２０（図４参照）において実行される。

【０２４４】

まず、暗号化対象のデータ格納セクタ位置をｓ（０≦ｓ≦３１（セクタ数３２の場合））とする（Ｓ１６０１）。次にそのセクタが暗号化対象であるかをチェック（Ｓ１６０２）する。このチェックは、セキュリティヘッダ（図７参照）の暗号化フラグ（Encryption Flag）に基づいて判定される。暗号化対象でない場合は、暗号化処理は実行されず、処理は終了する。暗号化対象である場合は、暗号化フォーマットタイプをチェック（Ｓ１６０３）する。これはセキュリティヘッダ内の暗号化フォーマットタイプ（Encryption Format Type）の設定をチェックするものであり、図８で説明したコンテンツ全体を１つの暗号化態様としているか、各セクタに異なる鍵を用いた暗号化処理を行なっているかを判定する。

【０２４５】

暗号化フォーマットタイプ（Encryption Format Type）の設定値が０の場合は、コンテンツ全体を１つの暗号化態様としている場合である。この場合は、ステップＳ１６０４において、暗号化アルゴリズム（Encryption Algorithm）の判定を行なう。暗号化アルゴリズムは、シングルＤＥＳかトリプルＤＥＳ（図２８参照）かを設定しているものであり、シングルＤＥＳであると判定された場合は、１つのコンテンツキーＫｃ（０）を適用して暗号化コンテンツの暗号化処理を実

行（Ｓ１６０６）する。トリプルＤＥＳであると判定された場合は、２つのコンテンツキーＫｃ（０）、Ｋｃ（１）を適用して暗号化コンテンツの暗号化処理を実行（Ｓ１６０７）する。

【０２４６】

一方、ステップＳ１６０３で、暗号フォーマットタイプ（Encryption Format Type）の設定値が１の場合は、各セクタに異なる鍵を用いた暗号化処理を行なう場合である。この場合は、ステップＳ１６０５において、暗号化アルゴリズム（Encryption Algorithm）の判定を行なう。暗号化アルゴリズムは、シングルＤＥＳかトリプルＤＥＳ（図２８参照）かを設定しているものであり、シングルＤＥＳであると判定された場合は、各セクタ（ｓ）に対応して設定されたコンテンツキーＫｃ（ｓ）を各セクタに適用して暗号化コンテンツの暗号化処理を実行（Ｓ１６０８）する。トリプルＤＥＳであると判定された場合は、２つのコンテンツキーＫｃ（ｓ）、Ｋｃ（ｓ＋１ｍｏｄ３２）を適用して各セクタ毎の暗号化処理を実行（Ｓ１６０９）する。

【０２４７】

セクタデータの復号処理の異なる処理態様を図４６に示す。図４６において、ステップＳ１７０１～Ｓ１７０８は、図４５の各ステップＳ１６０１～Ｓ１６０８と同様である。ステップＳ１７０９～Ｓ１７１１が図４５とは異なる。

【０２４８】

ステップＳ１７０５において、暗号化アルゴリズムがトリプルＤＥＳであると判定された場合、ステップＳ１７０９においてセクタＮｏ．（ｓ）を判定し、ｓが奇数である場合は、ｓ＝ｓ－１の更新を実行（Ｓ１７１０）し、各セクタに適用する鍵をＫｃ（ｓ），Ｋｃ（ｓ＋１）としてトリプルＤＥＳによる復号処理（Ｓ１７１１）を実行する。

【０２４９】

図４２に戻り、ファイルの暗号化書き込み処理フローの説明を続ける。上述の処理によってデータ部の暗号化処理ステップ（Ｓ１３３４）が終了すると、データ部に対する誤り訂正符号を生成（Ｓ１３３５）し、暗号化されたデータＤ（ｉ）とセクタデータに対応する改竄チェック値ＩＣＶと、誤り訂正符号を持つ冗長

部をメディアに書き込み（Ｓ１３３６）、書き込み成功フラグを１（成功）にセット（Ｓ１３３７）し、ビジーフラグを０（待機）に設定（Ｓ１３３９）する。

【０２５０】

なお、書き込み対象データがＢＰＴによる管理のなされていない内部メモリ内への書き込み処理である場合は、ステップＳ１３２９，Ｓ１３３０はスキップする。ステップＳ１３２９、Ｓ１３３０の判定がＮｏである場合、すなわちメディアのセットフラグが１でない場合、または、ＢＰＴにセクタＳ（ｉ）の書き込み許可が設定されていない場合には、ステップＳ１３３８に進み、書き込みエラーとして書き込み成功フラグを０にセットする。

【０２５１】

また、制御部は、ステップＳ１３４１～Ｓ１３４５において、メモリインタフェースのステータスを読み出して、ビジーフラグが０の状態において、書き込み成功フラグが１であることを条件としてアドレスを順次インクリメントして、書き込みデータを順次メモリインタフェースに送信する。すべての処理が終了すると、ファイル割り当てテーブルの更新処理を実行（Ｓ１３４６）し、更新したファイル割り当てテーブルを更新コマンドとともにメモリインタフェースに送信（Ｓ１３４７）し、メモリインタフェースはコマンドに従ってファイル割り当てテーブルの書き込み処理を実行（Ｓ１３４０）する。

【０２５２】

以上の、図４１～図４６によって説明した処理により、データの暗号化、メディアに対する格納処理が実行される。

【０２５３】

[リボケーションリストの更新]

次に、不正なメディアやコンテンツの失効情報としてのリボケーションリストの更新処理について説明する。前述したように、本発明におけるリボケーションリストは、複数の種類（ｅｘ．メディア、コンテンツ）の識別子（ＩＤ）から構成される。コンテンツやメディアの失効情報であるリボケーションリスト（Revocation List）に複数の種類のＩＤを設け，それぞれの照合を異なる動作として行うことによって、１つのリボケーションリストで複数の種類のコンテンツ、メ

ディアを排除することが可能となる。メディアの挿入時やコンテンツの読み出し時にメモリ・インターフェース部において、利用メディアまたは利用コンテンツの識別子（ＩＤ）と、リボケーションリストのリスティングＩＤとの照合を実行することにより、不正なメディアの使用や不正なコンテンツの読み出しを禁止することができる。

【０２５４】

先に説明したように、リボケーションリストには、リボケーションリストバージョン（Revocation List Version）が設定され、新たな不正なメディアやコンテンツの失効情報を追加した場合等にリボケーションリストは更新される。

【０２５５】

リボケーションリストの更新処理フローを図４７に示す。図４７において、左側はデバイスの制御部、右側はデバイスのメモリインタフェースである。

【０２５６】

まず、制御部は更新用のリボケーションリストを通信部２０１（図２参照）から受信する（Ｓ１８０１）と、更新用リボケーションリストチェックコマンドと、受信した更新用リボケーションリストをメモリインタフェースに送信（Ｓ１８０２）する。

【０２５７】

メモリインタフェースは、更新用リボケーションリストチェックコマンドと、更新用リボケーションリストを制御部から受信（Ｓ１８０３）すると、ビジーフラグを１（ビジー）に設定（Ｓ１８０４）し、リボケーションリストの改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ＿ｒｌを生成（Ｓ１８０５）する。

【０２５８】

リボケーションリストの改竄チェック用の改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｒｌは、予めデバイス内に格納されたリボケーションリスト（Revocation List）のＩＣＶ鍵を生成するマスター鍵：ＭＫicv_rlと、リボケーションリスト（Revocation List）のＩＣＶ鍵を生成する時の初期値：ＩＶicv_rlと、リボケーションリストの属性情報中に含まれるリボケーションリスト・バージョン（Version）に基づいて生成する。具体的には、改竄チェック値（ＩＣＶ）生成鍵

Ｋｉｃｖ_ｒｌ＝ＤＥＳ（Ｅ，ＭＫicv_rl，Version'ＩＶicv_rl）に基づいて改竄チェック値（ＩＣＶ）生成鍵が生成される。前記式の意味は、バージョン（Version）と初期値（ＩＶicv_rl）の排他論理和にマスター鍵：ＭＫicv_rlによるＤＥＳモードでの暗号化処理を実行するという意味である。

【０２５９】

次にメモリインタフェースは生成した改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｒｌを用いてリボケーションリストのＩＣＶ'を生成（Ｓ１８０６）し、予めリボケーションリスト内に格納された正しいＩＣＶ値と照合ＩＣＶ'＝ＩＣＶ？を実行（Ｓ１８０７）する。なお、ＩＣＶ'の生成処理は、前述の図１４で説明したＤＥＳモードに基づいて、初期値ＩＶrlを用い、生成した改竄チェック値（ＩＣＶ）生成鍵Ｋｉｃｖ_ｒｌを適用した処理によって行われる。

【０２６０】

ＩＣＶ'＝ＩＣＶである場合（Ｓ１８０７でＹｅｓ）は、更新用リボケーションリストが改竄のない正当なものであると判定され、ステップＳ１８０８に進み、現在セットされているリボケーションリストのバージョン（ｉ）と更新用リボケーションリストのバージョン（ｊ）を比較（Ｓ１８０９）し、更新用リボケーションリストのバージョンが新しい場合には、更新用リボケーションリストの有効フラグを１に設定（Ｓ１８１０）し、ビジーフラグを０にセット（Ｓ１８１１）して処理を終了する。

【０２６１】

一方、制御部側は、ステータス読み出しコマンドをメモリインタフェースに送信（Ｓ１８１２）し、ビジーフラグが０となった（Ｓ１８１３）ことを確認し、更新用リボケーションリスト有効フラグが１（Ｓ１８１４）である場合に、更新用リボケーションリストを通常のファイルとして内部メモリに保存（Ｓ１８１５）する。コンテンツの処理、メディアの装着時のチェックの際には、内部メモリに格納されたリボケーションリストが読み出される。

【０２６２】

以上、特定の実施例を参照しながら、本発明について詳解してきた。しかしながら、本発明の要旨を逸脱しない範囲で当業者が該実施例の修正や代用を成し得

ることは自明である。すなわち、例示という形態で本発明を開示してきたのであり、限定的に解釈されるべきではない。本発明の要旨を判断するためには、冒頭に記載した特許請求の範囲の欄を参酌すべきである。

【０２６３】

【発明の効果】

以上、説明したように、本発明のデータ記憶装置、およびデータ記録方法、データ再生方法によれば、コンテンツのセクタ対応の暗号処理鍵について、データ記録再生装置とデータ記憶装置間におけるＣＢＣモードでの暗号化を実行し、ＣＢＣモードで暗号化された鍵データをコンテンツに対応するヘッダに格納する構成としたので、暗号化された鍵データからの不正復号処理が困難となり、安全な鍵の保存が可能となる。

【０２６４】

さらに、本発明のデータ記憶装置、およびデータ記録方法、データ再生方法によれば、ＣＢＣモードによる鍵の暗号化処理をコンテンツを格納するメディアに固有の保存鍵を適用して行なう構成とし、例えばコンテンツ利用時には、相互認証の成立したメディアにおいて鍵データを復号することによってのみ鍵を取得可能としたので、デバイス単独、あるいは不正なメディアを装着した場合には鍵データの復号が不可能であり安全な鍵の保存が可能となる。

【図面の簡単な説明】

【図１】

本発明のデータ処理装置の使用概念を説明する図である。

【図２】

本発明のデータ処理装置のデバイスおよびメディアの構成を示す図である。

【図３】

本発明のデータ処理装置のメモリ格納データ構成を示す図である。

【図４】

本発明のデータ処理装置にデバイスのメモリインタフェースの詳細構成を示す図である。

【図５】

本発明のデータ処理装置におけるメモリインタフェースのステータスレジスタのデータ構成を示す図である。

【図６】

本発明のデータ処理装置におけるメディアに格納されるデータの詳細構成を示す図である。

【図７】

本発明のデータ処理装置においてメディアに格納されるコンテンツに対応して設定されるセキュリティヘッダの構成を説明する図である。

【図８】

本発明のデータ処理装置におけるデータ暗号化の２つの態様を説明する図である。

【図９】

本発明のデータ処理装置におけるリボケーションリストの構成を示す図である。

【図１０】

本発明のデータ処理装置におけるブロック・パーミッション・テーブル（ＢＰＴ）について説明する図である。

【図１１】

本発明のデータ処理装置におけるメディア１製造時のＢＰＴ格納処理フローを示す図である。

【図１２】

本発明のデータ処理装置におけるメディア２製造時のＢＰＴ格納処理フローを示す図である。

【図１３】

本発明のデータ処理装置におけるブロック・パーミッション・テーブル（ＢＰＴ）の具体例について説明する図である。

【図１４】

本発明のデータ処理装置における改竄チェック値生成処理構成について説明する図である。

【図１５】

本発明のデータ処理装置における改竄チェック値検証処理フローについて説明する図である。

【図１６】

本発明のデータ処理装置におけるデバイス起動時フローを示す図である。

【図１７】

本発明のデータ処理装置におけるファイル割り当てテーブルの構成例について説明する図である。

【図１８】

本発明のデータ処理装置におけるメディア１認識時フロー（その１）を示す図である。

【図１９】

本発明のデータ処理装置におけるメディア１認識時フロー（その２）を示す図である。

【図２０】

本発明のデータ処理装置におけるメディア２認識時フロー（その１）を示す図である。

【図２１】

本発明のデータ処理装置におけるメディア２認識時フロー（その２）を示す図である。

【図２２】

本発明のデータ処理装置においてデバイス・メディア間において実行される相互認証処理シーケンスを示す図である。

【図２３】

本発明のデータ処理装置における相互認証・鍵共有処理フロー（その１）を示す図である。

【図２４】

本発明のデータ処理装置における相互認証・鍵共有処理フロー（その２）を示す図である。

【図２５】

本発明のデータ処理装置におけるファイルの読み出し処理フローを示す図である。

【図２６】

本発明のデータ処理装置におけるファイルの書き込み処理フローを示す図である。

【図２７】

本発明のデータ処理装置におけるメモリに格納されたデータの暗号化処理態様を説明する図である。

【図２８】

本発明のデータ処理装置におけるメモリに格納されたデータの暗号化処理態様として適用可能なトリプルＤＥＳを説明する図である。

【図２９】

本発明のデータ処理装置におけるメモリに格納されたデータの暗号化処理態様を説明する図である。

【図３０】

本発明のデータ処理装置におけるメモリに格納されたデータの暗号化処理態様を説明する図である。

【図３１】

本発明のデータ処理装置におけるセクタ対応改竄チェック値の格納処理態様を説明する図である。

【図３２】

本発明のデータ処理装置におけるセクタ対応コンテンツキー他の鍵の暗号化処理例を説明する図である。

【図３３】

本発明のデータ処理装置におけるセクタ対応コンテンツキー他の鍵の復号処理例を説明する図である。

【図３４】

本発明のデータ処理装置におけるセクタ対応コンテンツキー他の鍵のデバイス

・メディア間における処理例を説明する図である。

【図３５】

本発明のデータ処理装置におけるファイルの復号読み出し処理フロー（その１）を示す図である。

【図３６】

本発明のデータ処理装置におけるファイルの復号読み出し処理フロー（その２）を示す図である。

【図３７】

本発明のデータ処理装置におけるコンテンツキー他の復号処理フローを示す図である。

【図３８】

本発明のデータ処理装置におけるコンテンツキー他のメディアの保存鍵による復号処理フローを示す図である。

【図３９】

本発明のデータ処理装置におけるセクタデータの復号処理フロー（その１）を示す図である。

【図４０】

本発明のデータ処理装置におけるセクタデータの復号処理フロー（その２）を示す図である。

【図４１】

本発明のデータ処理装置におけるファイルの暗号化書き込み処理フロー（その１）を示す図である。

【図４２】

本発明のデータ処理装置におけるファイルの暗号化書き込み処理フロー（その２）を示す図である。

【図４３】

本発明のデータ処理装置におけるコンテンツキー他の暗号化処理フローを示す図である。

【図４４】

本発明のデータ処理装置におけるコンテンツキー他のメディアの保存鍵による暗号化処理フローを示す図である。

【図４５】

本発明のデータ処理装置におけるセクタデータの暗号化処理フロー（その１）を示す図である。

【図４６】

本発明のデータ処理装置におけるセクタデータの暗号化処理フロー（その２）を示す図である。

【図４７】

本発明のデータ処理装置におけるリボケーションリストの更新処理フローを示す図である。

【符号の説明】

１０１　システム運営者
１０２　デバイス
１０３　メデイア
２００　デバイス
２０１　通信部
２０２　入力部
２０３　表示部
２０４　デバイスコントローラ
２０５　制御部
２０７　メモリ部
３００　メモリインタフェース（Ｉ／Ｆ）部
２１０　メディア１
２１１　制御部
２１２　メモリ部
２３０　メディア２
２３１　コントローラ
２３２　メモリ部

２３３　制御部

２３４　メモリインタフェース（Ｉ／Ｆ）部

２３５　内部メモリ

２３６　暗号処理部

３０１　ステータスレジスタ

３０２　コマンドレジスタ

３０３　アドレスレジスタ

３０４　カウントレジスタ

３０５　コントロールレジスタ

３０６　送受信制御部

３０７　送信バッファメモリ

３０８　受信バッファメモリ

３０９　送信レジスタ

３１０　受信レジスタ

３２０　暗号処理部

３２１　メモリ部

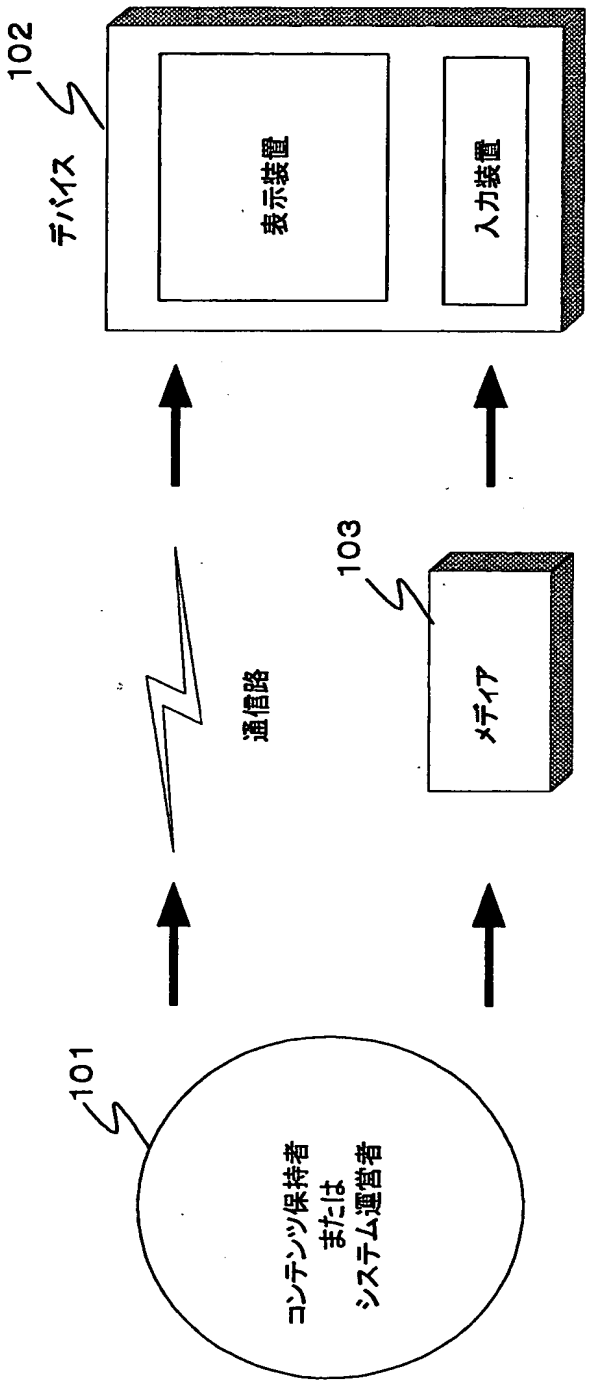
３２３　ＥＣＣ回路

３２４　外部メモリ入出力インタフェース

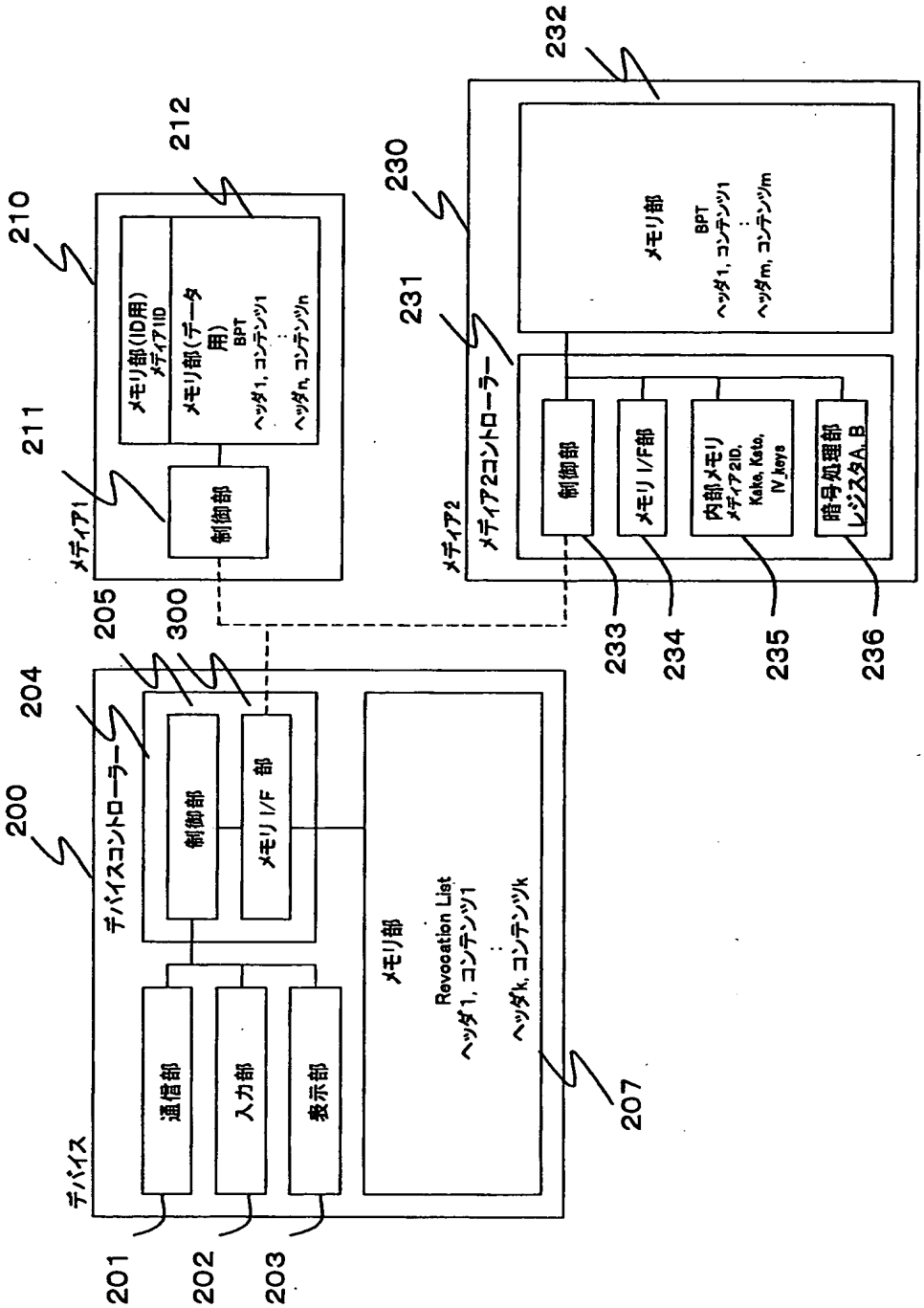
３２５　内部メモリ入出力インタフェース

【書類名】　　図面

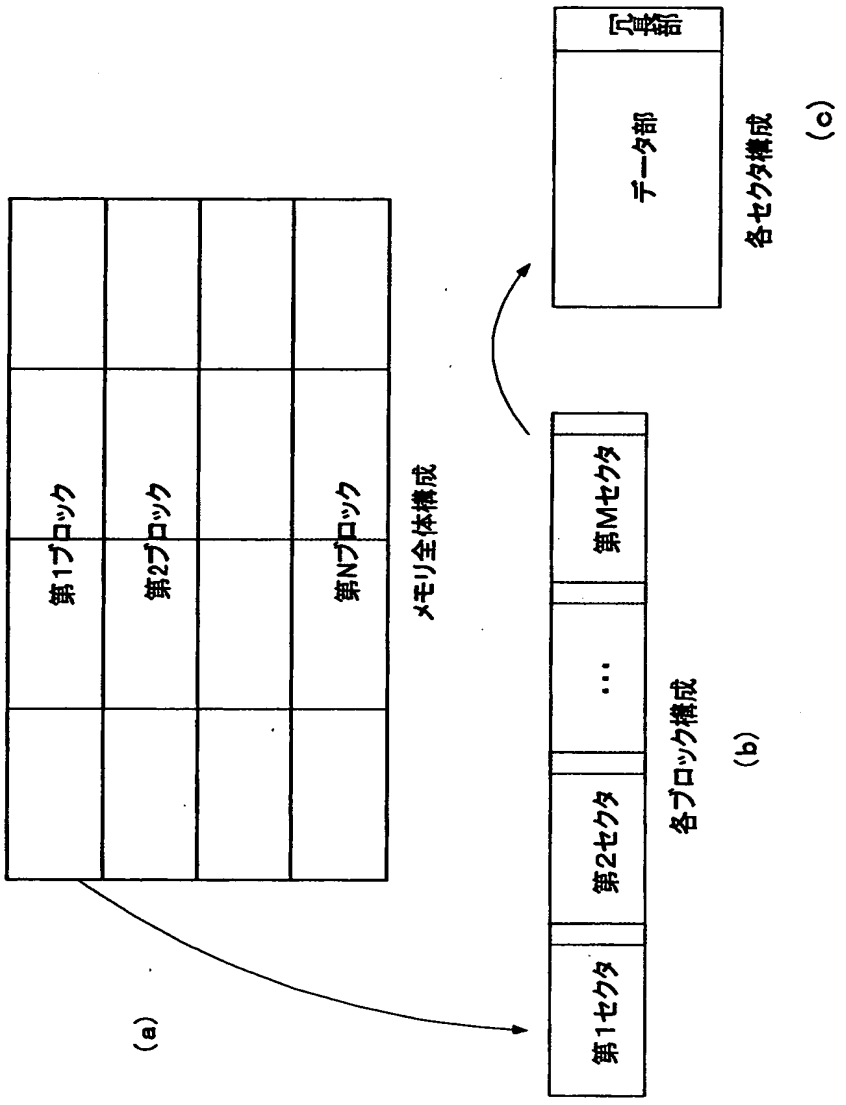
【図１】



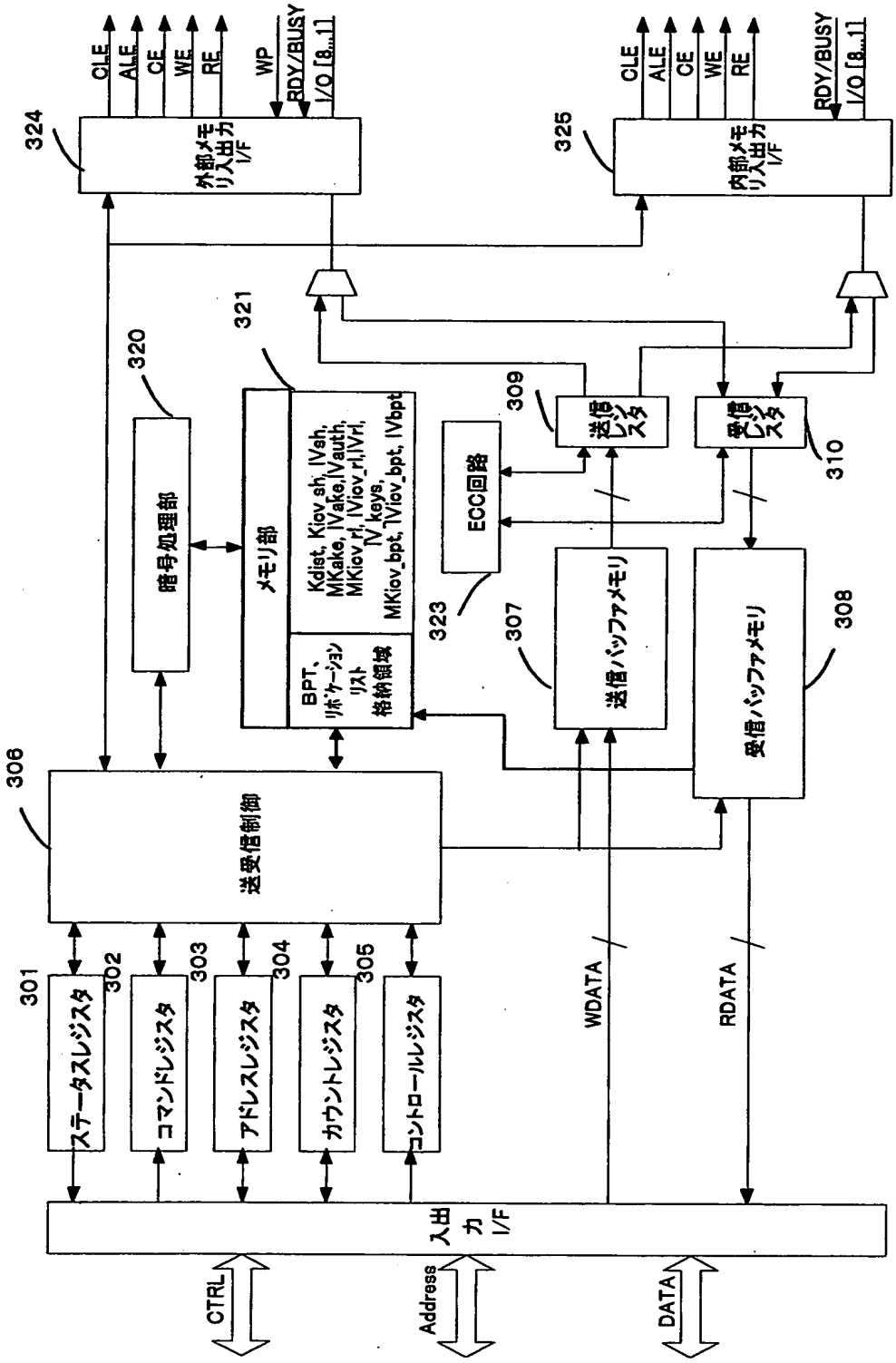
【図２】


200
デバイスコントローラー
204
制御部
メモリ I/F 部
205
300
通信部
入力部
表示部
メモリ部
Revocation List
ヘッダ1, コンテンツ1
ヘッダk, コンテンツk
207
デバイス
201
202
203
210
212
メモリ部(ID用)
メディアIID
メモリ部(データ用)
BPT
ヘッダ1, コンテンツ1
ヘッダn, コンテンツn
211
制御部
メディア1
230
232
メモリ部
BPT
ヘッダ1, コンテンツ1
ヘッダm, コンテンツm
231
制御部
メモリ I/F部
内部メモリ
メディアID,
Kaka, Kata,
IV_keys
暗号処理部
レジスタA, B
メディア2コントローラー
メディア2
233
234
235
236

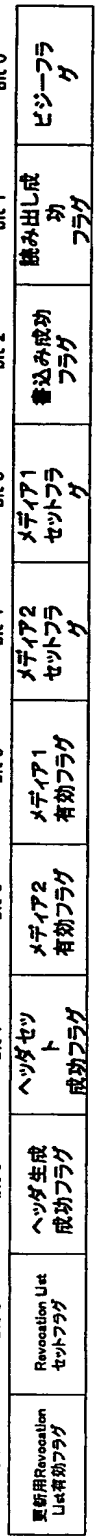
【図３】


第1ブロック
第2ブロック
第Nブロック
メモリ全体構成
(a)
第1セクタ
第2セクタ
…
第Mセクタ
各ブロック構成
(b)
データ部
冗長部
各セクタ構成
(c)

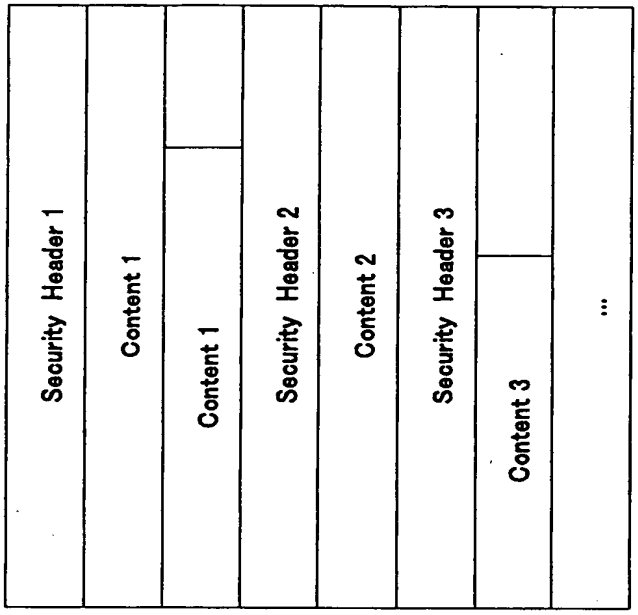
【図４】

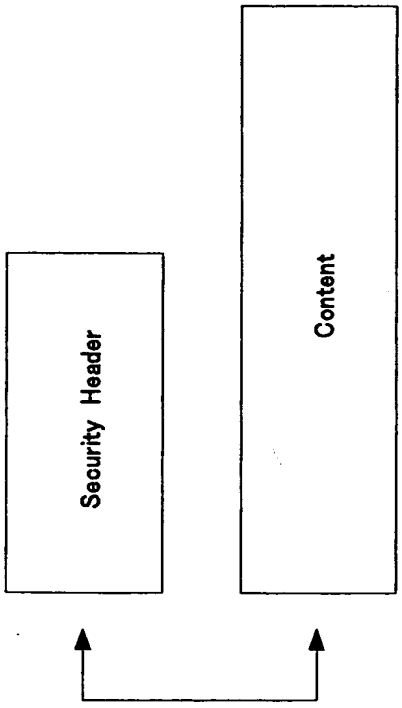

CLE
ALE
CE
WE
RE
WP
RDY/BUSY
I/O [8...1]
324
外部メモリ入出力 I/F
325
内部メモリ入出力 I/F
320
暗号処理部
321
メモリ部
Kdist, Kiov_sh, IVsh, MKake, IVake, IVauth, MKiov_rl, IViov_rl, IVrl, IV_keys, MKiov_bpt, IViov_bpt, IVbpt
BPT、リボケーションリスト格納領域
323
ECC回路
309
送信レジスタ
310
受信レジスタ
307
送信バッファメモリ
308
受信バッファメモリ
306
送受信制御
301
ステータスレジスタ
302
コマンドレジスタ
303
アドレスレジスタ
304
カウントレジスタ
305
コントロールレジスタ
WDATA
RDATA
入出力 I/F
CTRL
Address
DATA

【図５】

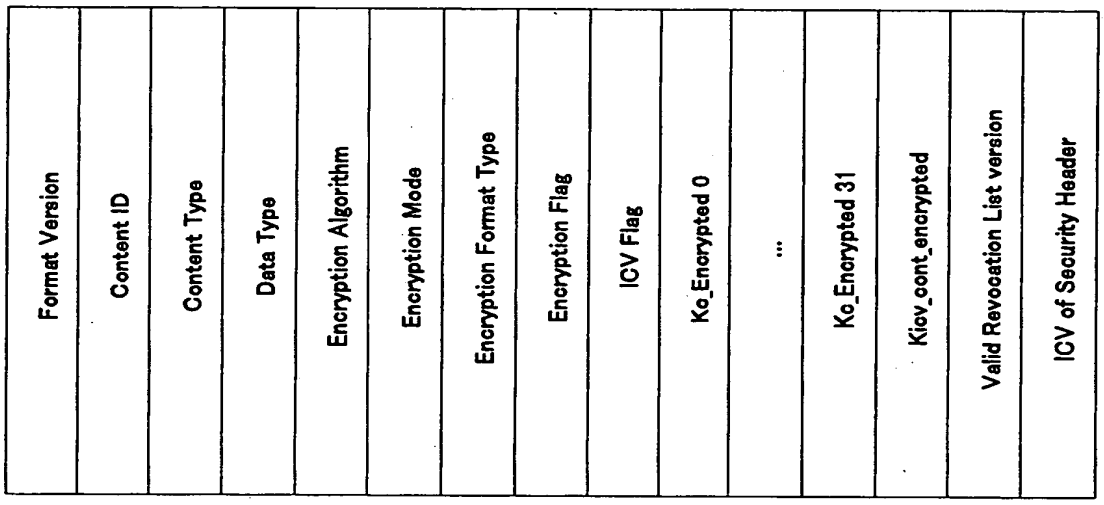

bit 10
bit 9
bit 8
bit 7
bit 6
bit 5
bit 4
bit 3
bit 2
bit 1
bit 0
更新用Revocation List有効フラグ
Revocation List セットフラグ
ヘッダ生成成功フラグ
ヘッダセット成功フラグ
メディア2有効フラグ
メディア1有効フラグ
メディア2セットフラグ
メディア1セットフラグ
書込み成功フラグ
読み出し成功フラグ
ビジーフラグ

【図６】

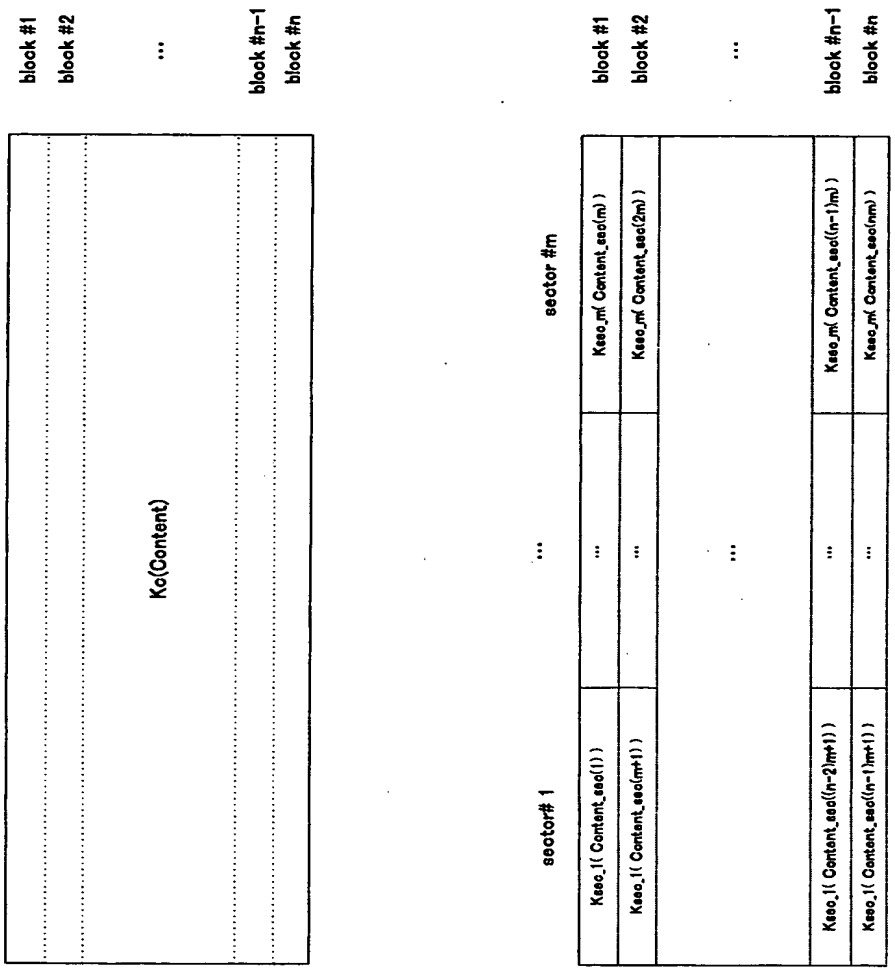

Security Header 1
Content 1
Content 1
Security Header 2
Content 2
Security Header 3
Content 3
...
Block #1
Block #2
Block #3
Block #4
Block #5
Block #6
Block #7
...
(b)


Security Header
Content
(a)

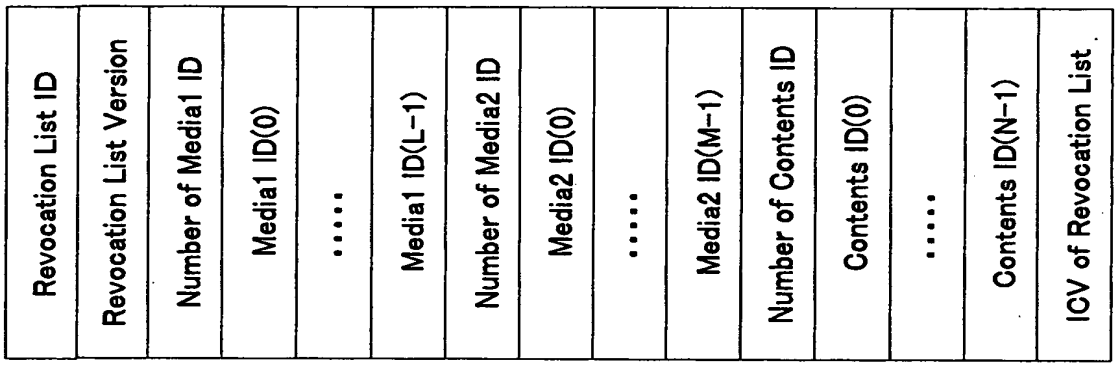
【図７】


Format Version
Content ID
Content Type
Data Type
Encryption Algorithm
Encryption Mode
Encryption Format Type
Encryption Flag
ICV Flag
Ko_Encrypted 0
...
Ko_Encrypted 31
Kiov_cont_encrypted
Valid Revocation List version
ICV of Security Header

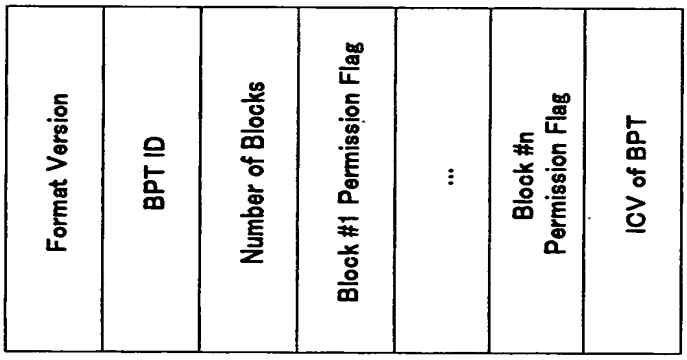
【図８】


block #1
block #2
...
block #n-1
block #n
Ko(Content)
sector# 1
...
sector #m
Kseo_1(Content_seo(1))
Kseo_1(Content_seo(m+1))
Kseo_1(Content_seo((n-2)m+1))
Kseo_1(Content_seo((n-1)m+1))
Kseo_m(Content_seo(m))
Kseo_m(Content_seo(2m))
Kseo_m(Content_seo((n-1)m))
Kseo_m(Content_seo(nm))
各ブロックのセクタ＃１は
Kseo_1 で暗号化
各ブロックのセクタ＃ｍは
Kseo_m で暗号化

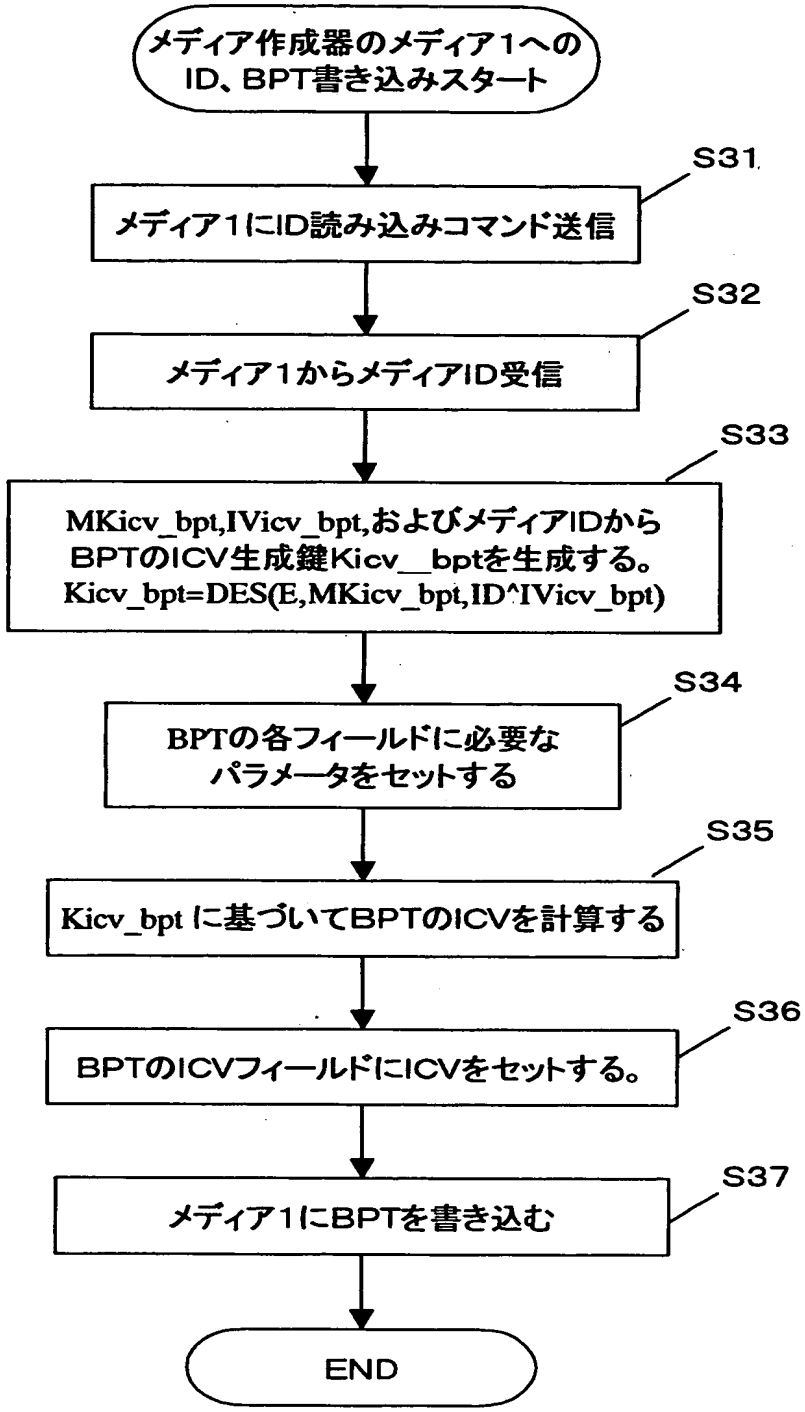
【図９】


Revocation List ID
Revocation List Version
Number of Media1 ID
Media1 ID(0)
.....
Media1 ID(L-1)
Number of Media2 ID
Media2 ID(0)
.....
Media2 ID(M-1)
Number of Contents ID
Contents ID(0)
.....
Contents ID(N-1)
ICV of Revocation List

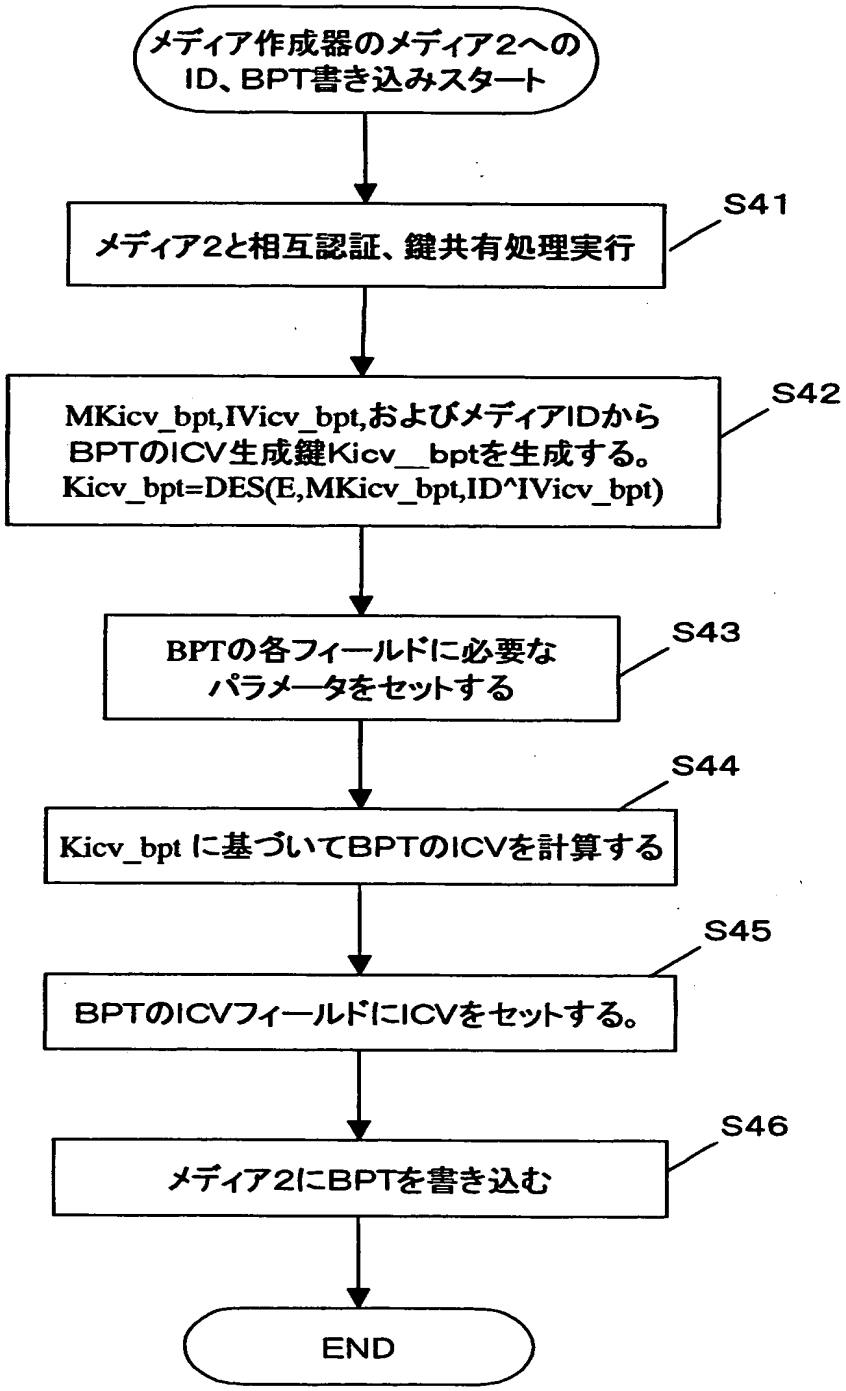
【図１０】


Format Version
BPT ID
Number of Blocks
Block #1 Permission Flag
...
Block #n Permission Flag
ICV of BPT

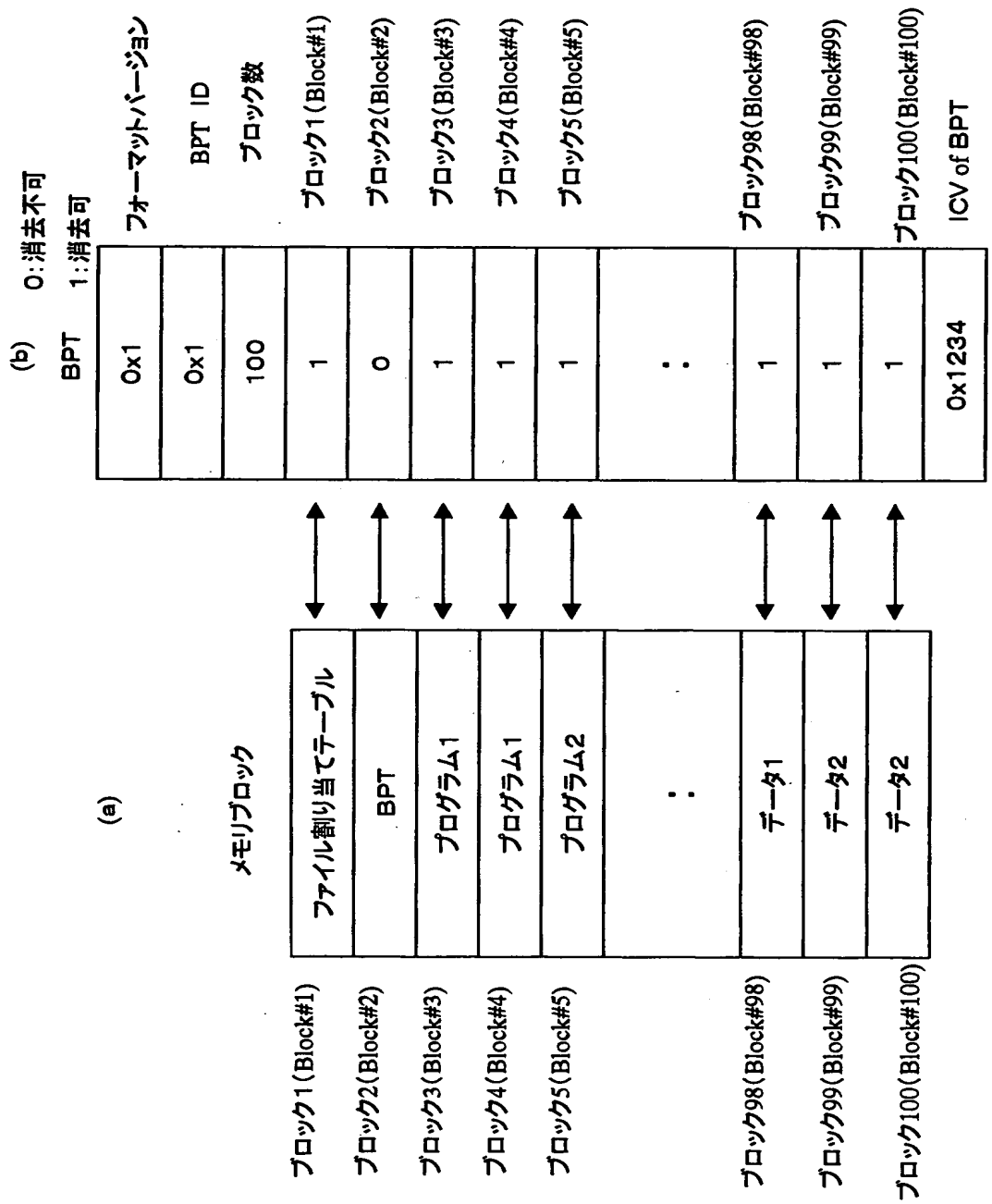
【図１１】


メディア作成器のメディア1への
ID、BPT書き込みスタート
S31
メディア1にID読み込みコマンド送信
S32
メディア1からメディアID受信
S33
MKicv_bpt,IVicv_bpt,およびメディアIDから
BPTのICV生成鍵Kicv__bptを生成する。
Kicv_bpt=DES(E,MKicv_bpt,ID^IVicv_bpt)
S34
BPTの各フィールドに必要な
パラメータをセットする
S35
Kicv_bpt に基づいてBPTのICVを計算する
S36
BPTのICVフィールドにICVをセットする。
S37
メディア1にBPTを書き込む
END

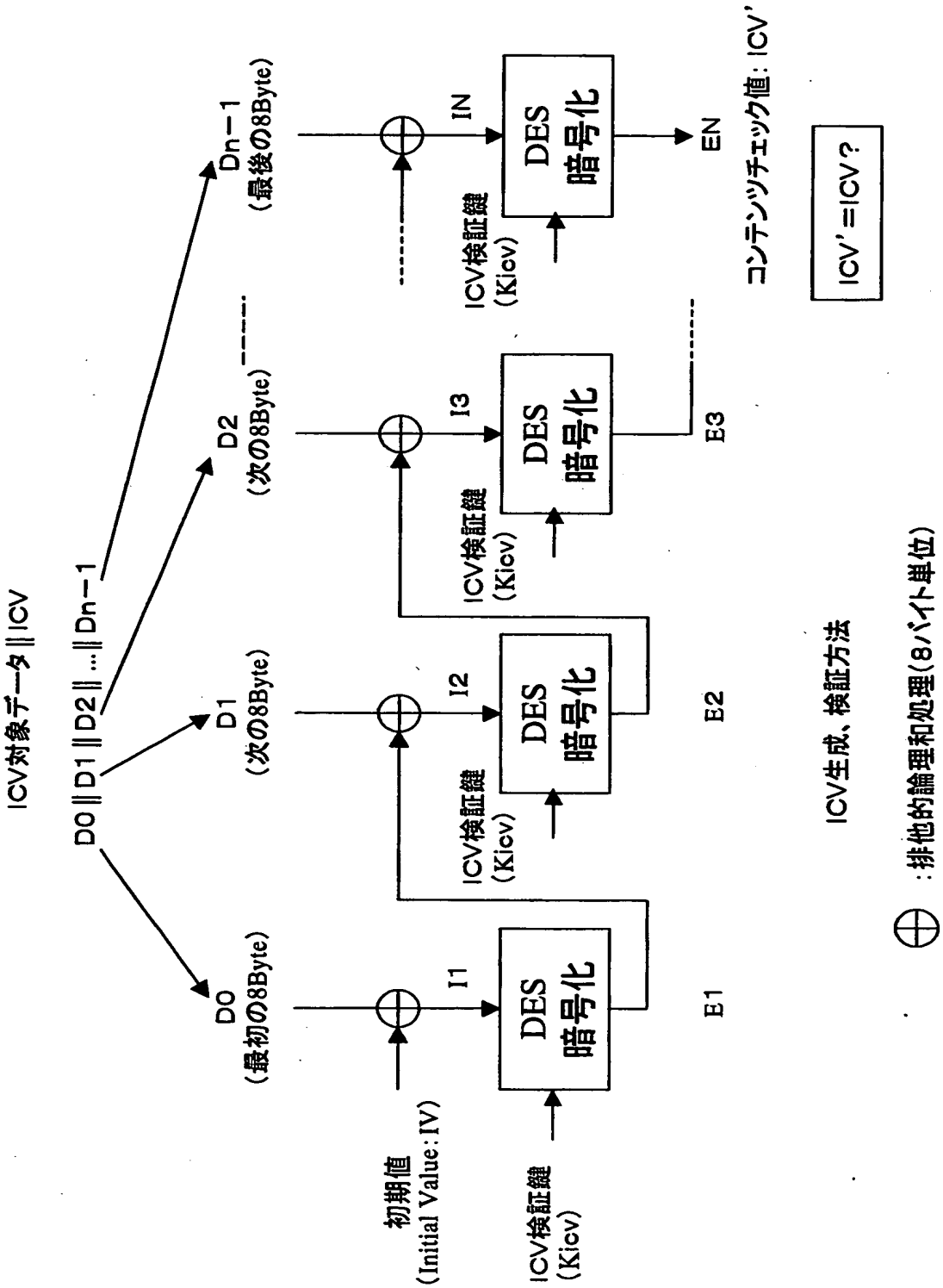
【図１２】


メディア作成器のメディア2へのID、BPT書き込みスタート
S41
メディア2と相互認証、鍵共有処理実行
S42
MKicv_bpt,IVicv_bpt,およびメディアIDからBPTのICV生成鍵Kicv__bptを生成する。Kicv_bpt=DES(E,MKicv_bpt,ID^IVicv_bpt)
S43
BPTの各フィールドに必要なパラメータをセットする
S44
Kicv_bpt に基づいてBPTのICVを計算する
S45
BPTのICVフィールドにICVをセットする。
S46
メディア2にBPTを書き込む
END

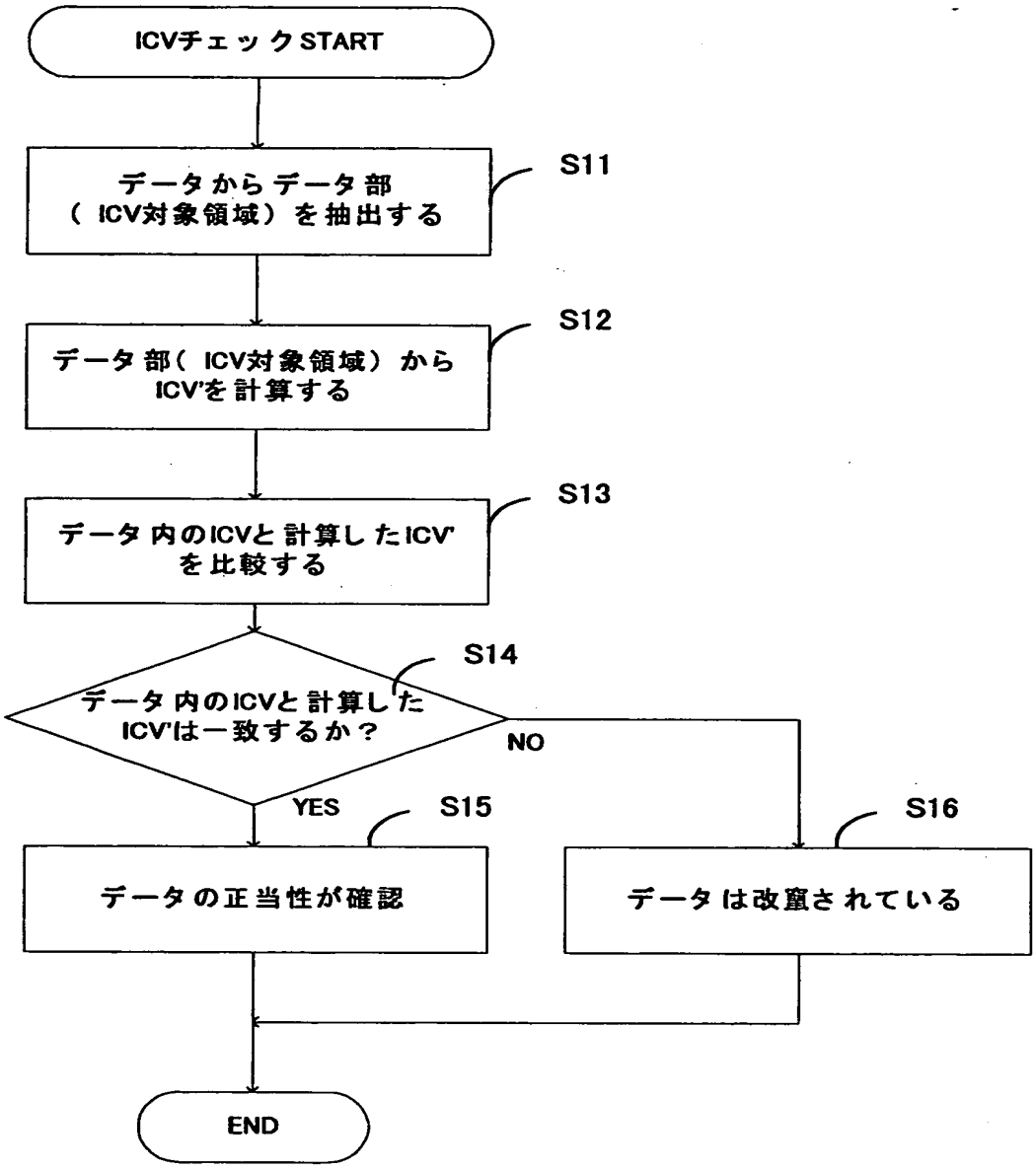
【図１３】


(a)
メモリブロック
ファイル割り当てテーブル
BPT
プログラム1
プログラム1
プログラム2
データ1
データ2
データ2
ブロック1(Block#1)
ブロック2(Block#2)
ブロック3(Block#3)
ブロック4(Block#4)
ブロック5(Block#5)
ブロック98(Block#98)
ブロック99(Block#99)
ブロック100(Block#100)
(b)
BPT 1:消去可 0:消去不可
フォーマットバージョン
BPT ID
ブロック数
ブロック1(Block#1)
ブロック2(Block#2)
ブロック3(Block#3)
ブロック4(Block#4)
ブロック5(Block#5)
ブロック98(Block#98)
ブロック99(Block#99)
ブロック100(Block#100)
ICV of BPT
0x1
0x1
100
1
0
1
1
1
1
1
1
0x1234

【図１４】

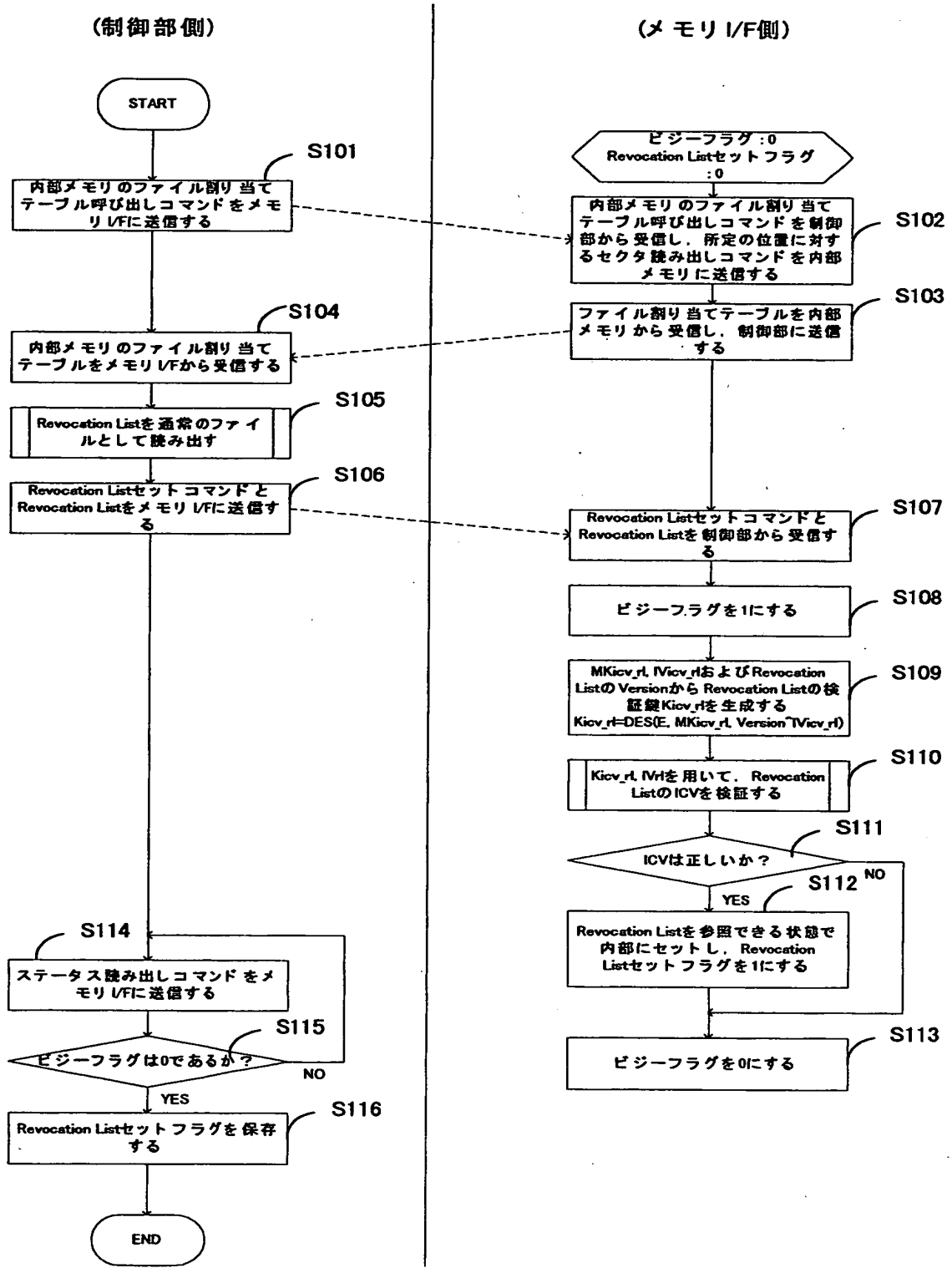

ICV対象データ‖ICV
D0‖D1‖D2‖...‖Dn−1
D0
(最初の8Byte)
D1
(次の8Byte)
D2
(次の8Byte)
Dn−1
(最後の8Byte)
初期値
(Initial Value: IV)
ICV検証鍵
(Kicv)
DES
暗号化
I1
I2
I3
IN
E1
E2
E3
EN
コンテンツチェック値: ICV'
ICV'=ICV?
ICV生成、検証方法
:排他的論理和処理(8バイト単位)

【図１５】


ICVチェックSTART
データからデータ部（ICV対象領域）を抽出する
S11
データ部（ICV対象領域）からICV'を計算する
S12
データ内のICVと計算したICV'を比較する
S13
データ内のICVと計算したICV'は一致するか？
S14
NO
YES
S15
S16
データの正当性が確認
データは改竄されている
END

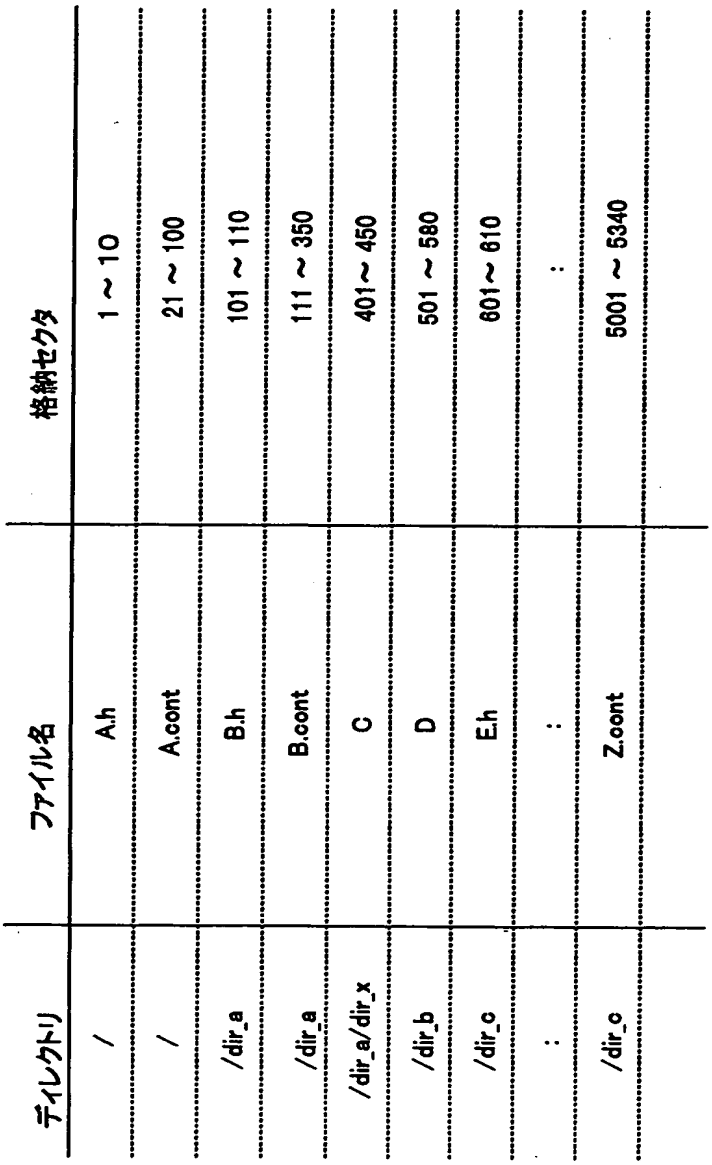
ICVチェック

【図１６】



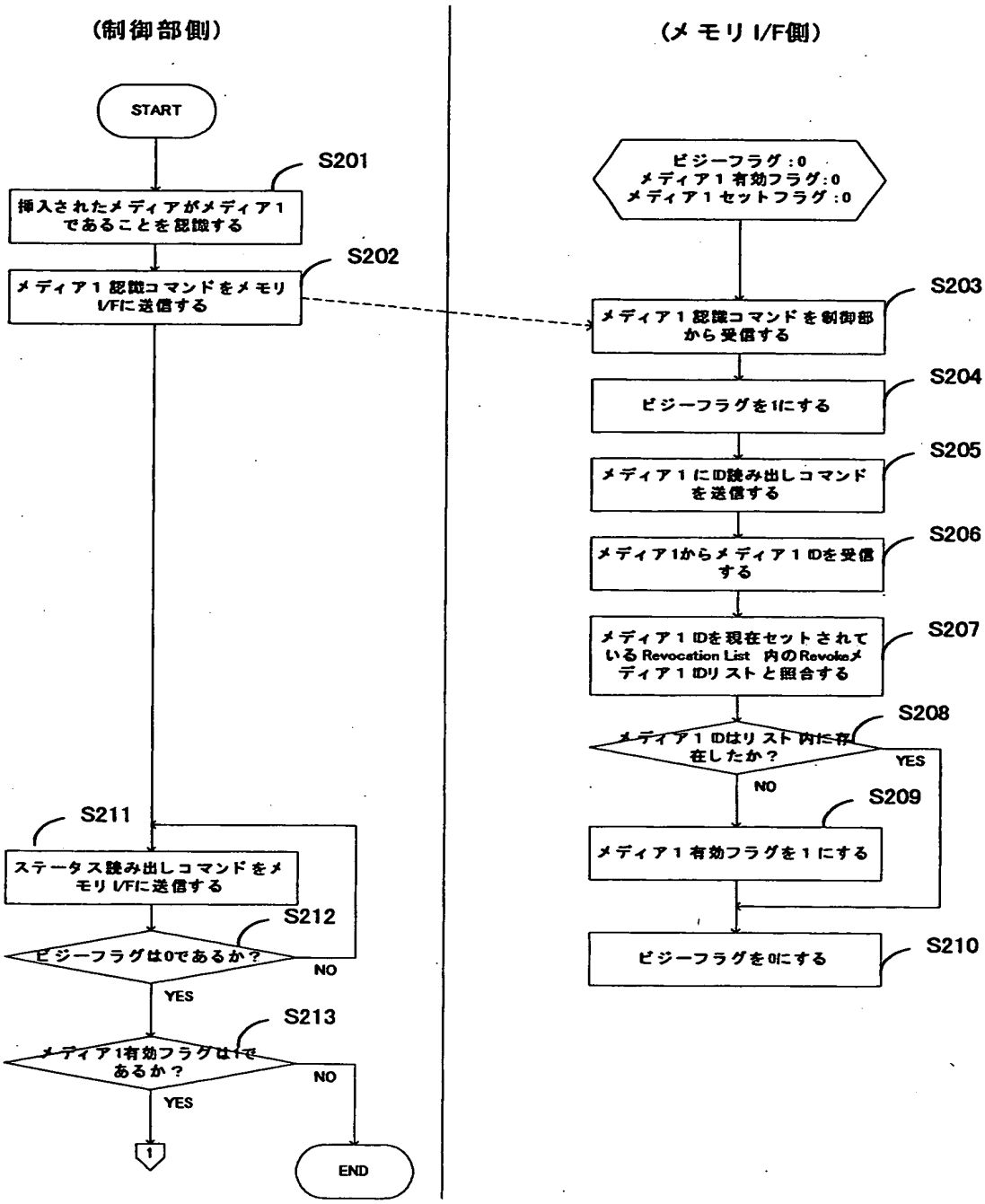
デバイス起動時フロー

【図１７】



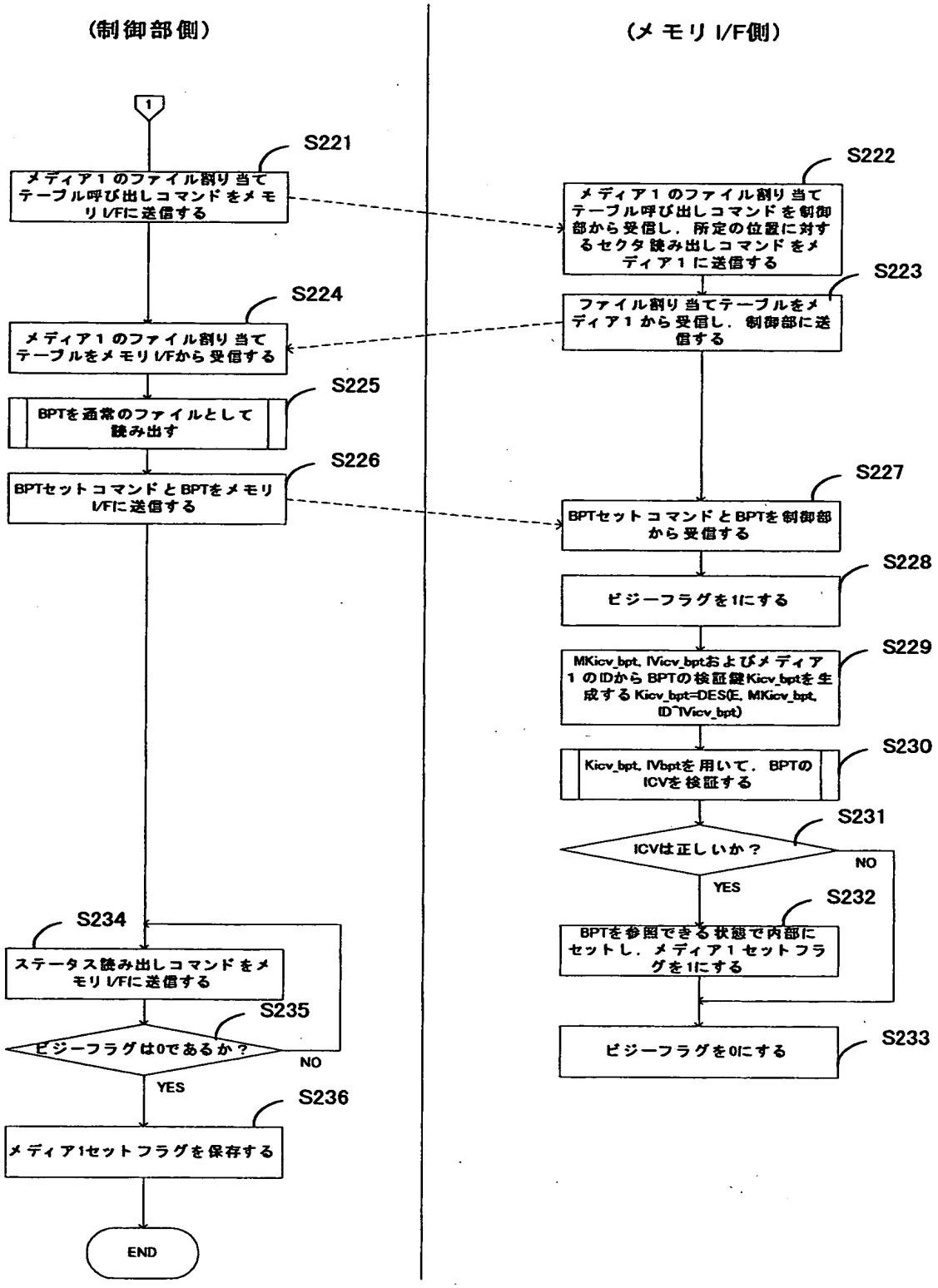
ディレクトリ	ファイル名	格納セクタ
/	A.h	1 ~ 10
/	A.cont	21 ~ 100
/dir_a	B.h	101 ~ 110
/dir_a	B.cont	111 ~ 350
/dir_a/dir_x	C	401~ 450
/dir_b	D	501 ~ 580
/dir_c	E.h	601~ 610
:	:	:
/dir_c	Z.cont	5001 ~ 5340

【図１８】



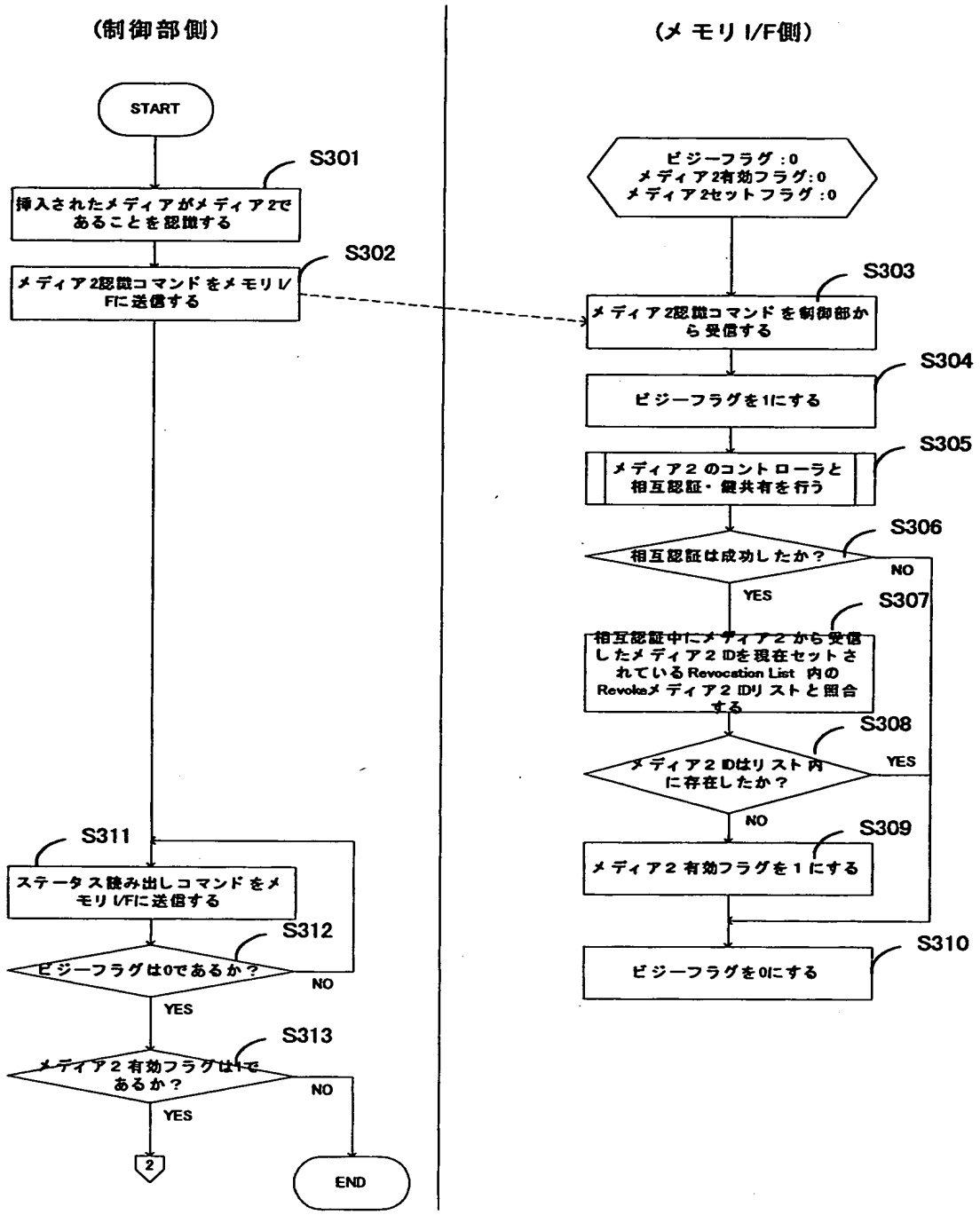
メディア１認識時フロー

【図１９】



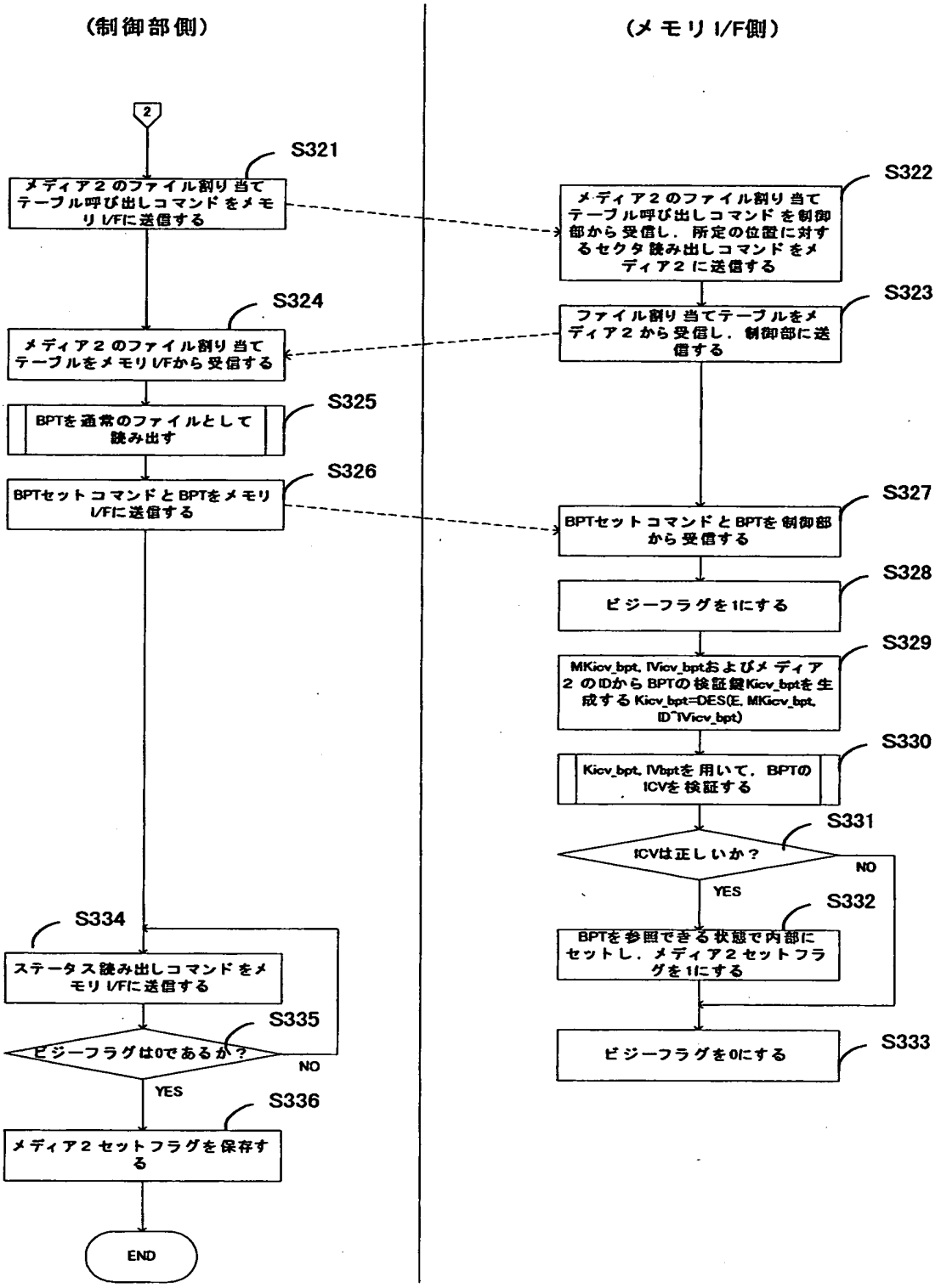
メディア１認識時フロー（cont.）

【図２０】



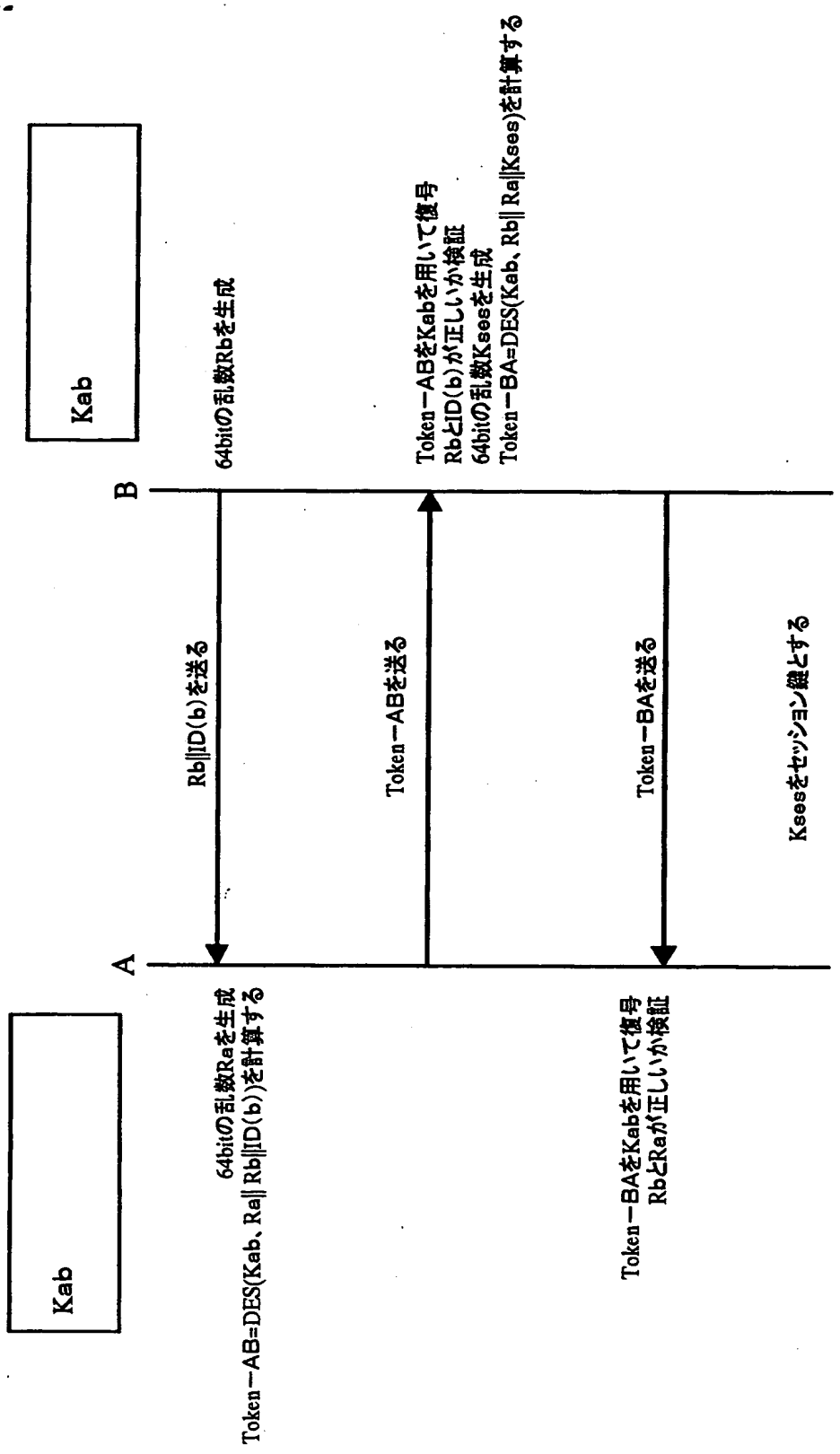
メディア２認識時フロー

【図２１】



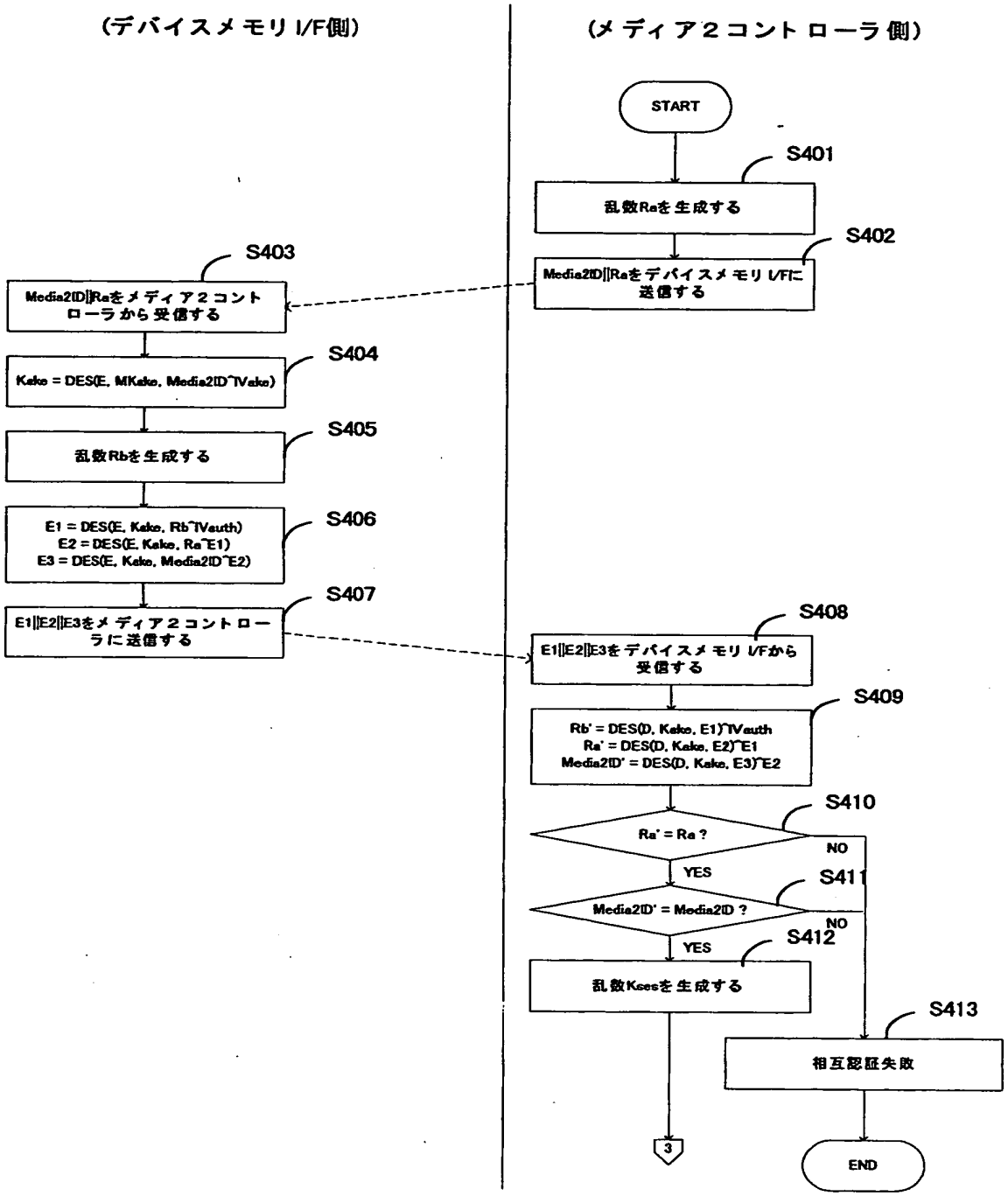
メディア２認識時フロー（cont.）

【図２２】



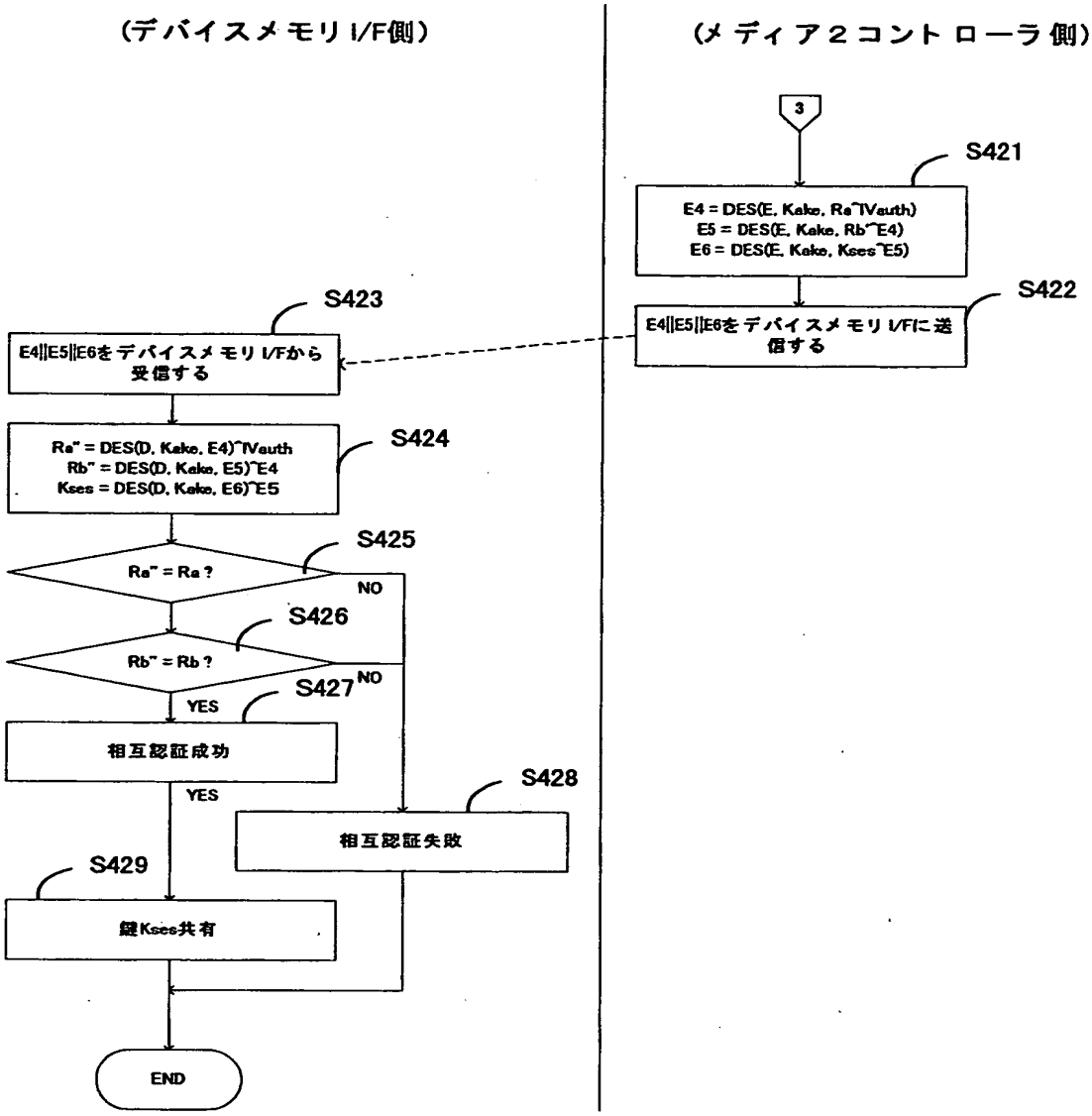
ISO/IEC 9798-2　対称鍵暗号技術を用いた相互認証および鍵共有方式

【図２３】



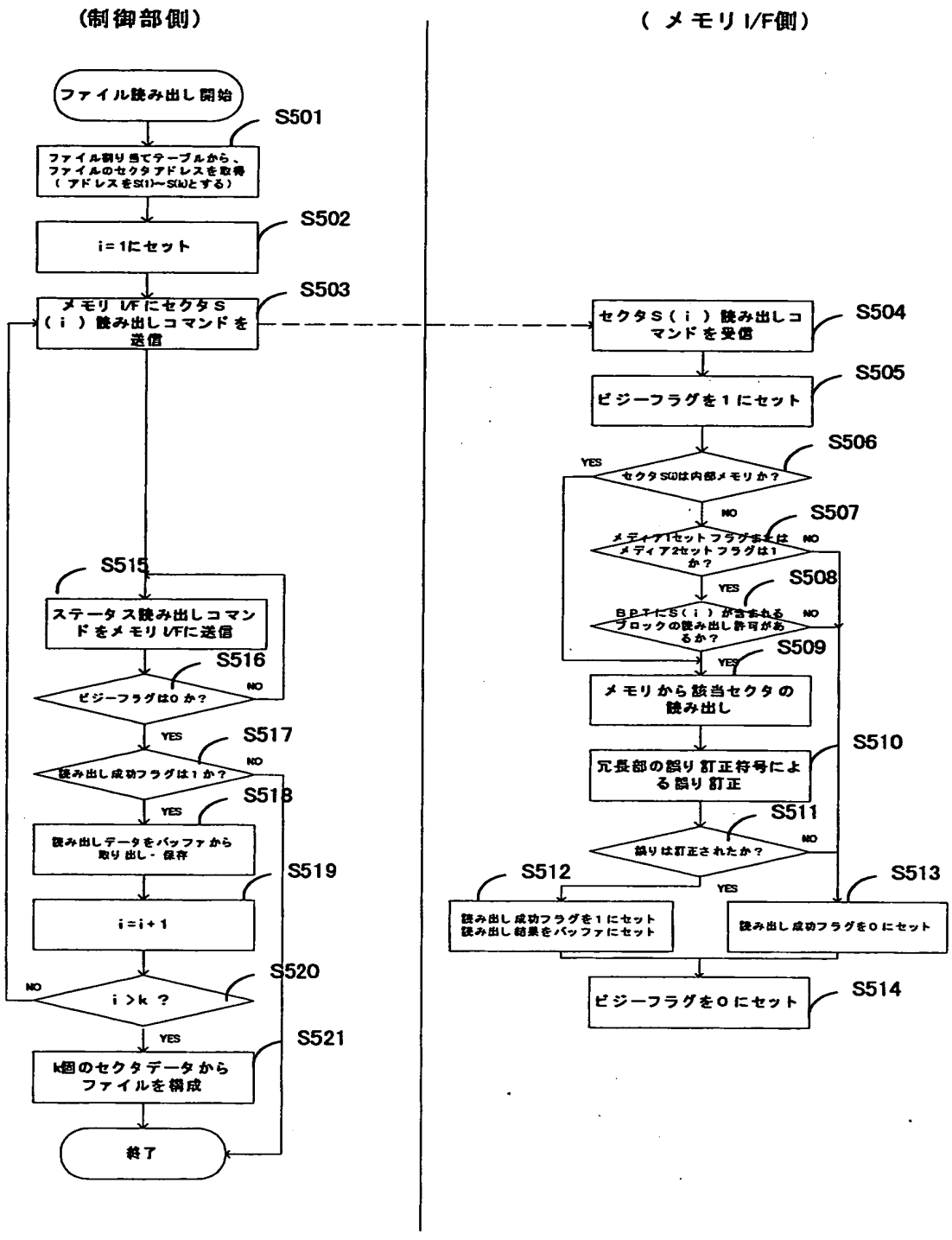
相互認証・鍵共有フロー

【図２４】



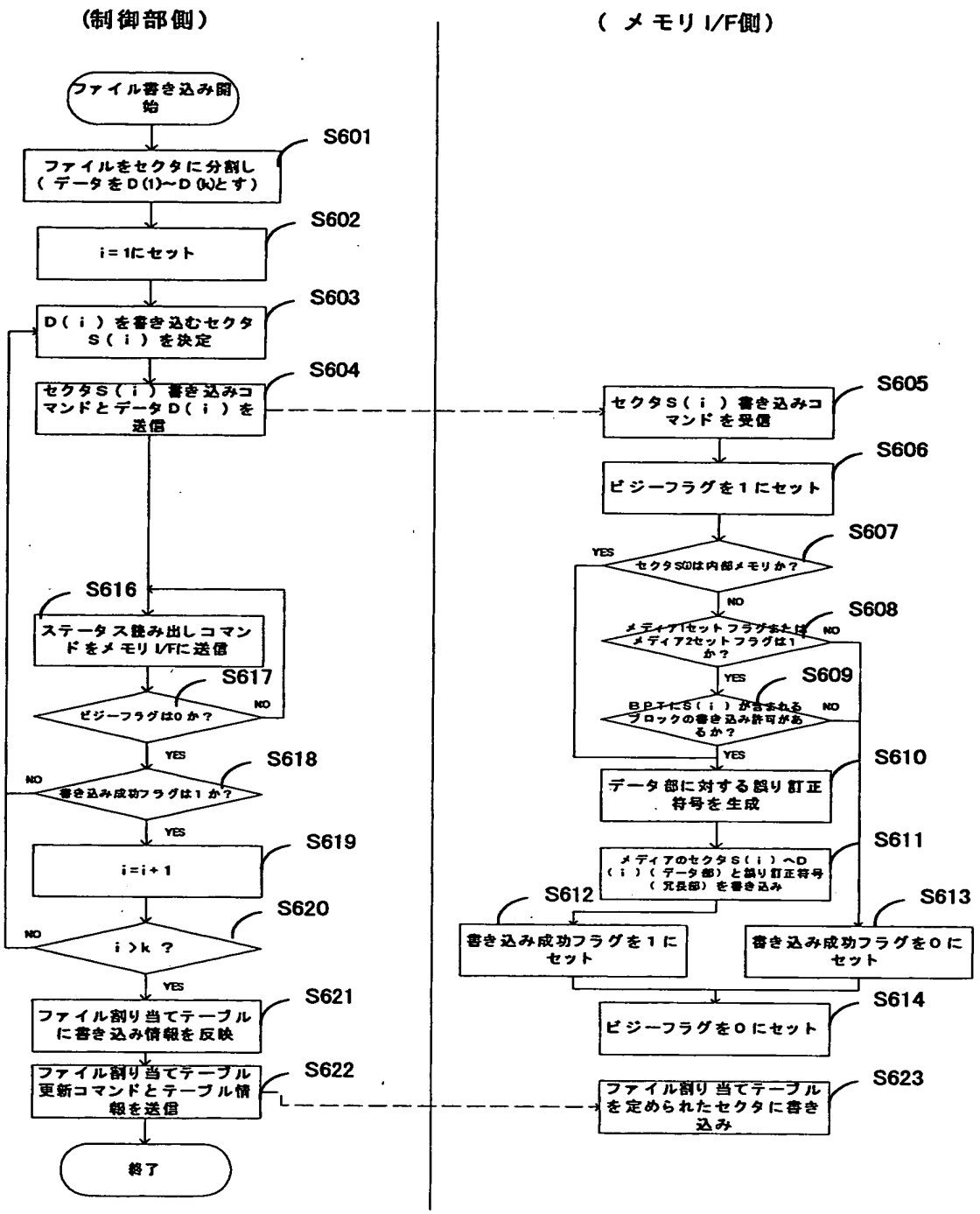
フロー 1-4：相互認証・鍵共有フロー（cont.）

【図２５】



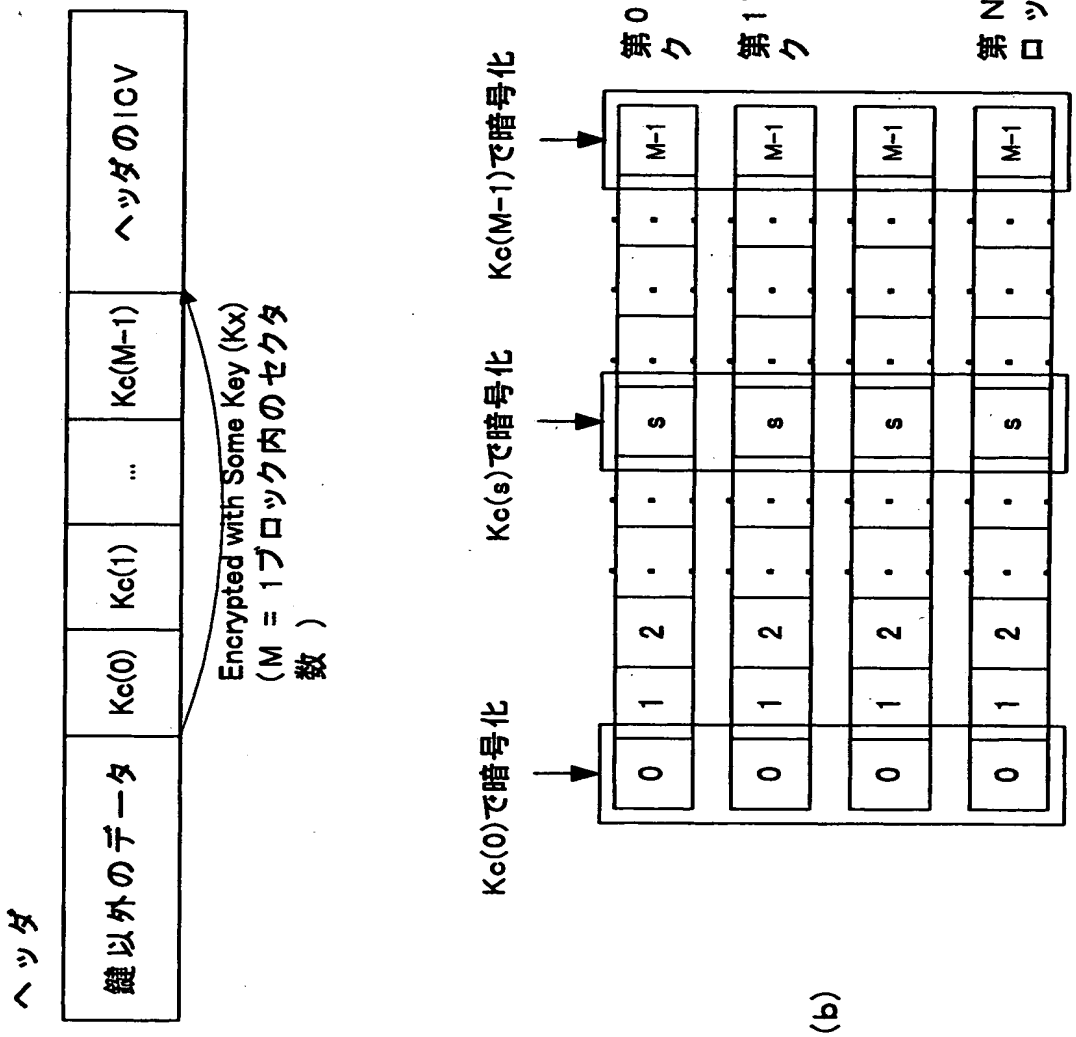
ファイルの読み出し処理

【図２６】

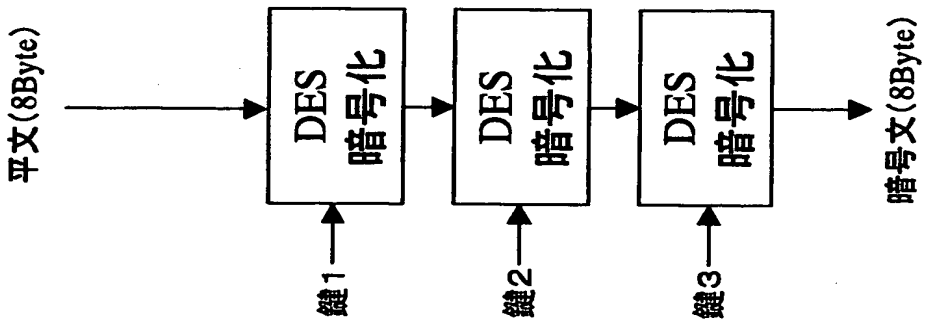


ファイルの書き込み処理

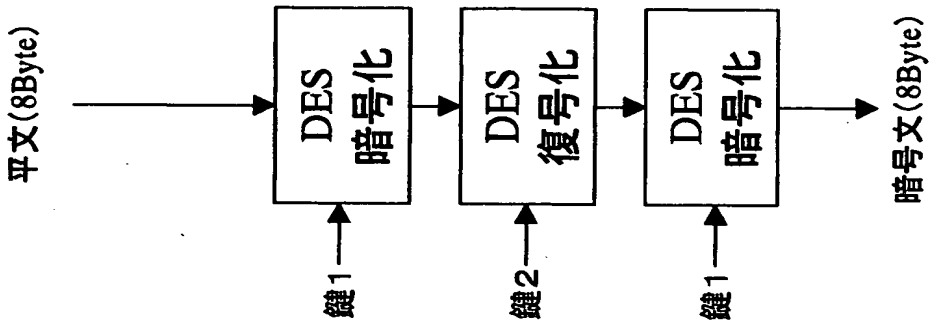
【図２７】


ヘッダ
鍵以外のデータ
Kc(0)
Kc(1)
...
Kc(M-1)
ヘッダのICV
Encrypted with Some Key (Kx)
（M＝1ブロック内のセクタ数）
(a)
Kc(0)で暗号化
Kc(s)で暗号化
Kc(M-1)で暗号化
第0ブロック
第1ブロック
第Nブロック
0
1
2
s
M-1
(b)

【図２８】

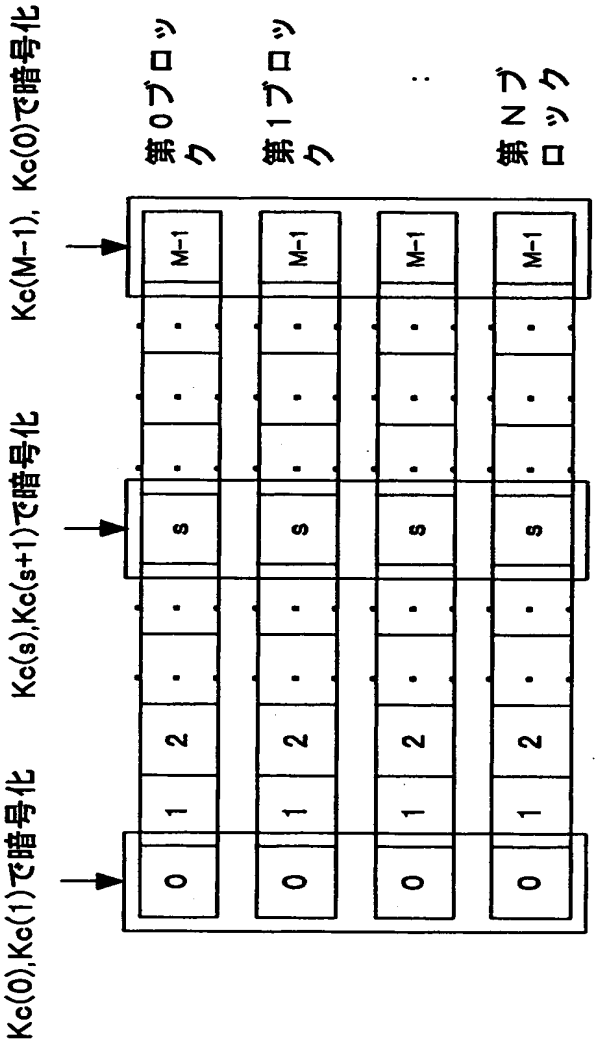

平文(8Byte)
DES
暗号化
DES
暗号化
DES
暗号化
暗号文(8Byte)
鍵1
鍵2
鍵3

(b)

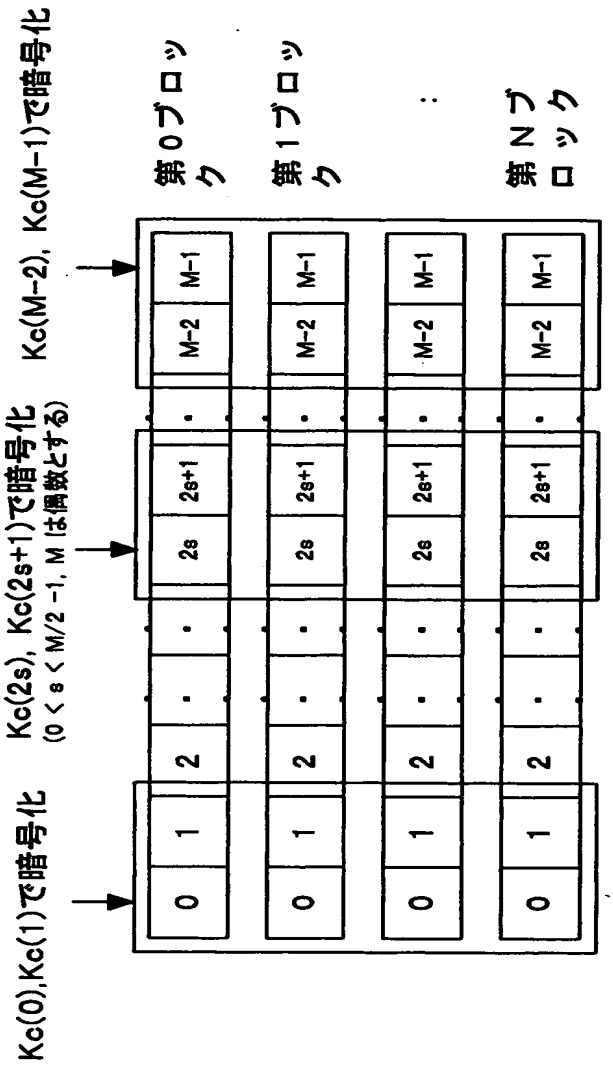

平文(8Byte)
DES
暗号化
DES
復号化
DES
暗号化
暗号文(8Byte)
鍵1
鍵2
鍵1

(a)

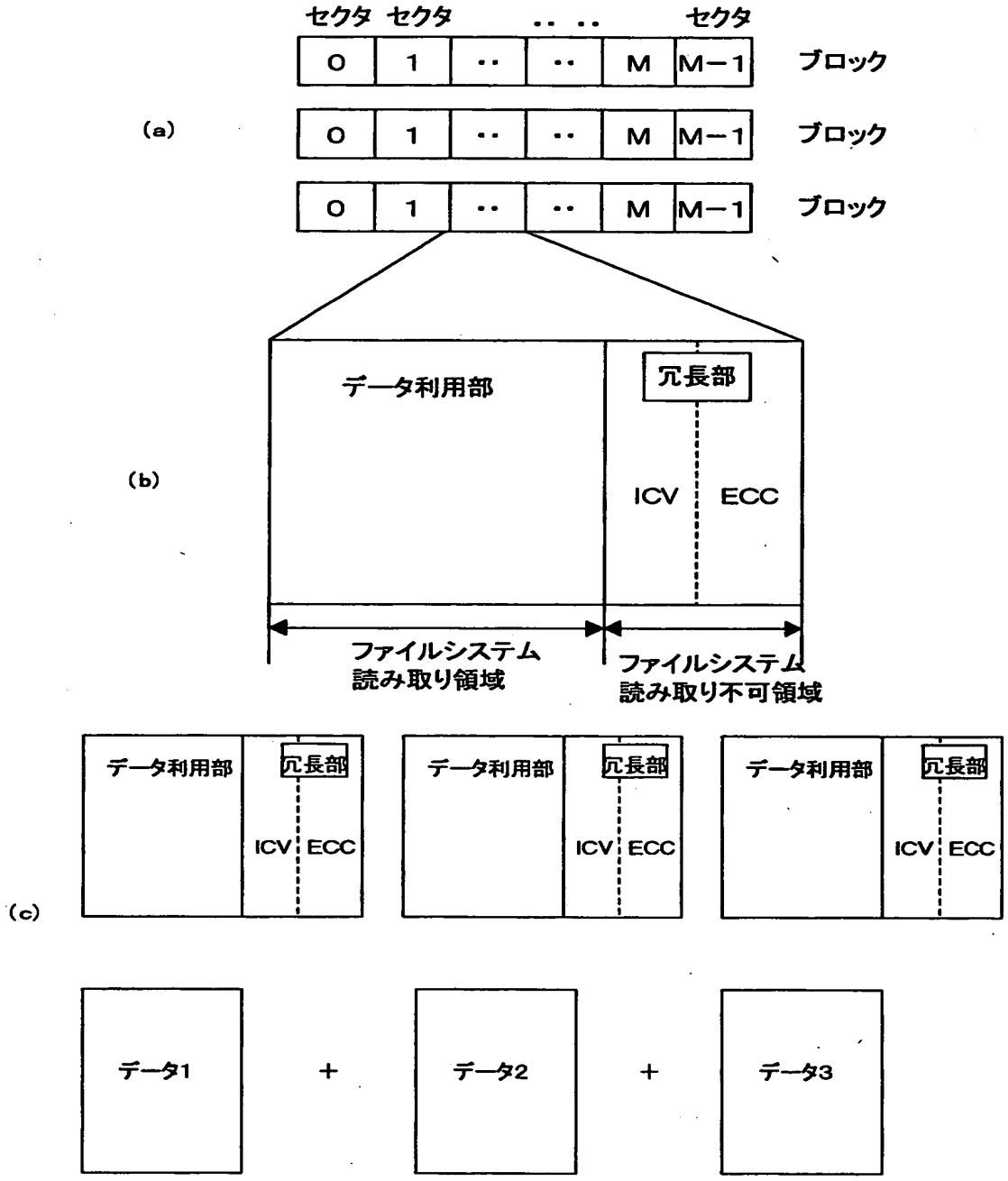
【図２９】


Kc(0),Kc(1)で暗号化
Kc(s),Kc(s+1)で暗号化
Kc(M−1), Kc(0)で暗号化
第０ブロック
第１ブロック
第Ｎブロック
0
1
2
s
M−1

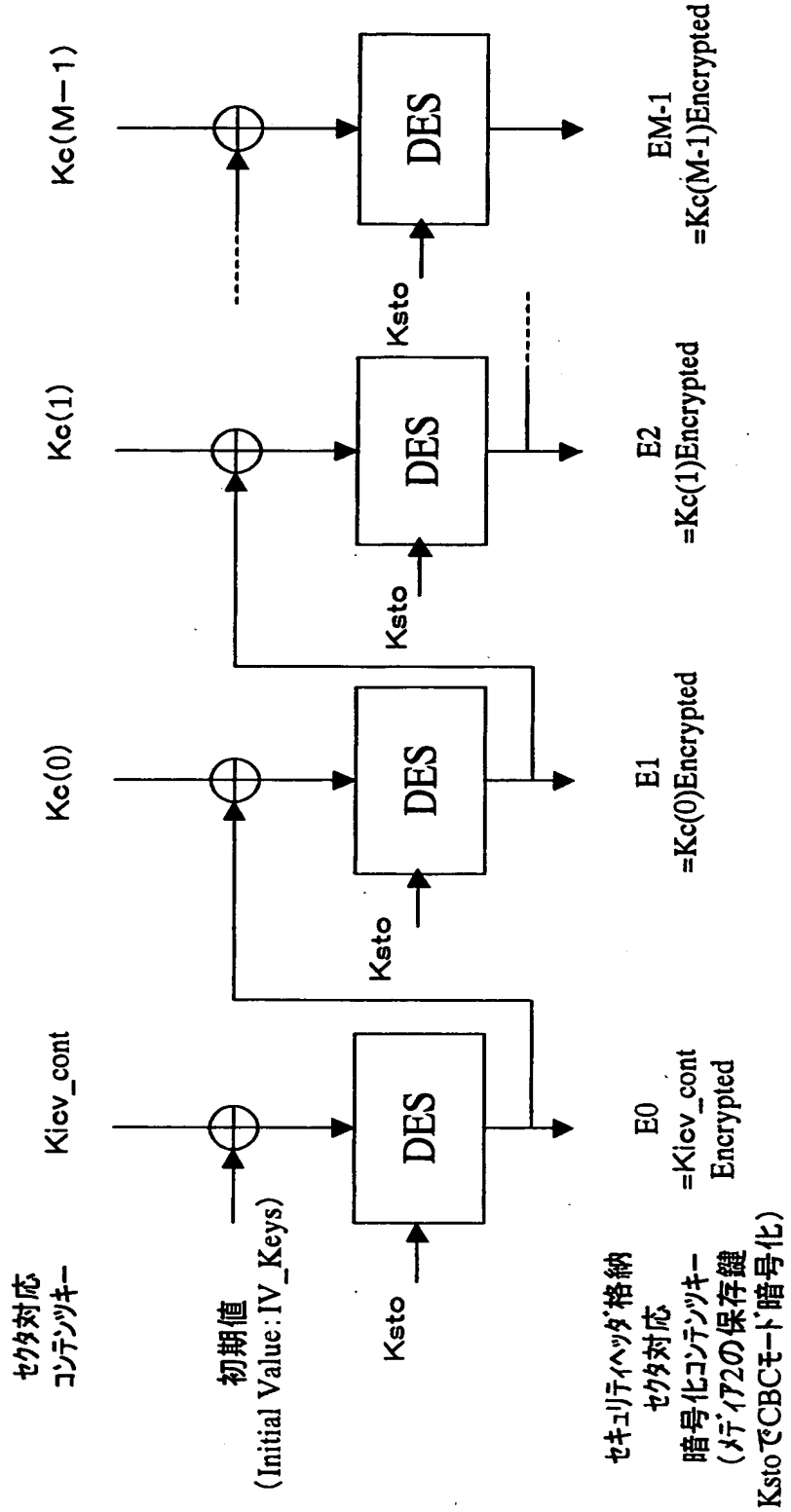
【図３０】


Kc(0),Kc(1)で暗号化
Kc(2s), Kc(2s+1)で暗号化
(0 < s < M/2 -1, M は偶数とする)
Kc(M-2), Kc(M-1)で暗号化
第0ブロック
第1ブロック
第Nブロック
0
1
2
2s
2s+1
M-2
M-1

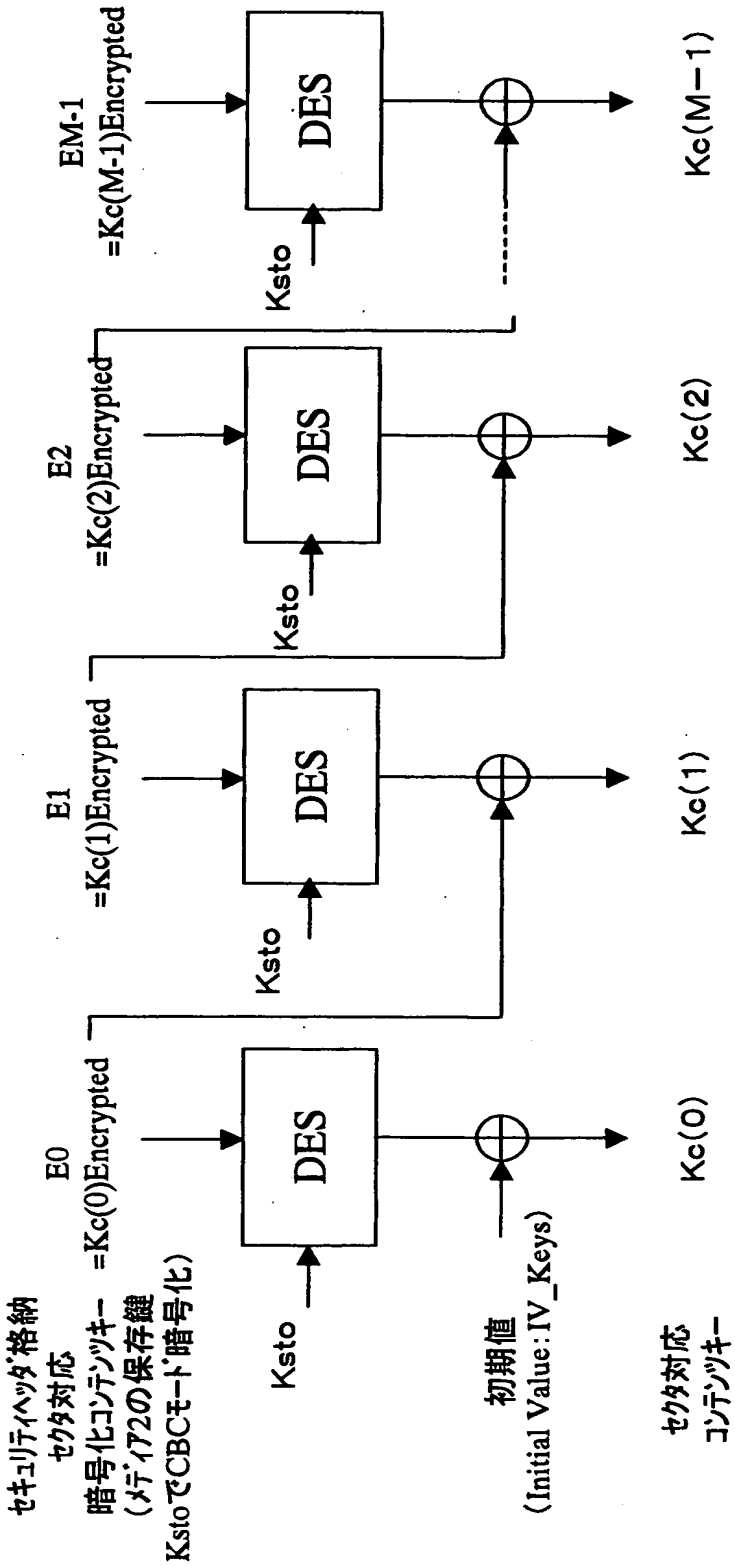
【図３１】


セクタ
セクタ
セクタ
0
1
M
M−1
ブロック
(a)
(b)
データ利用部
冗長部
ICV
ECC
ファイルシステム
読み取り領域
ファイルシステム
読み取り不可領域
(c)
データ1
データ2
データ3
+

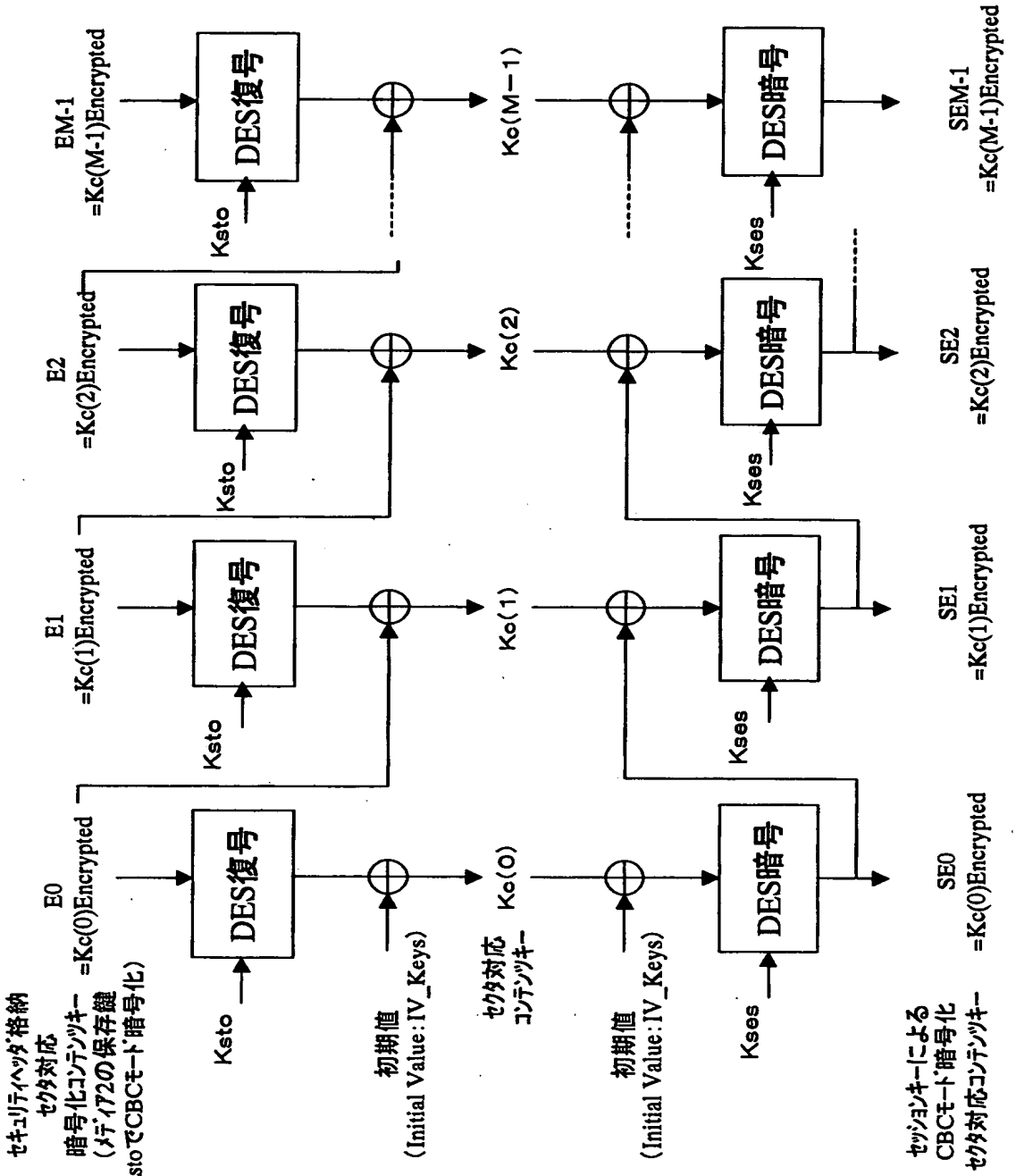
【図３２】


セクタ対応
コンテンツキー
Kicv_cont
初期値
(Initial Value:IV_Keys)
Ksto
DES
E0
=Kicv_cont
Encrypted
セキュリティヘッダ格納
セクタ対応
暗号化コンテンツキー
(メディア2の保存鍵
KstoでCBCモード暗号化)
Kc(0)
Ksto
DES
E1
=Kc(0)Encrypted
Kc(1)
Ksto
DES
E2
=Kc(1)Encrypted
Kc(M－1)
Ksto
DES
EM-1
=Kc(M-1)Encrypted

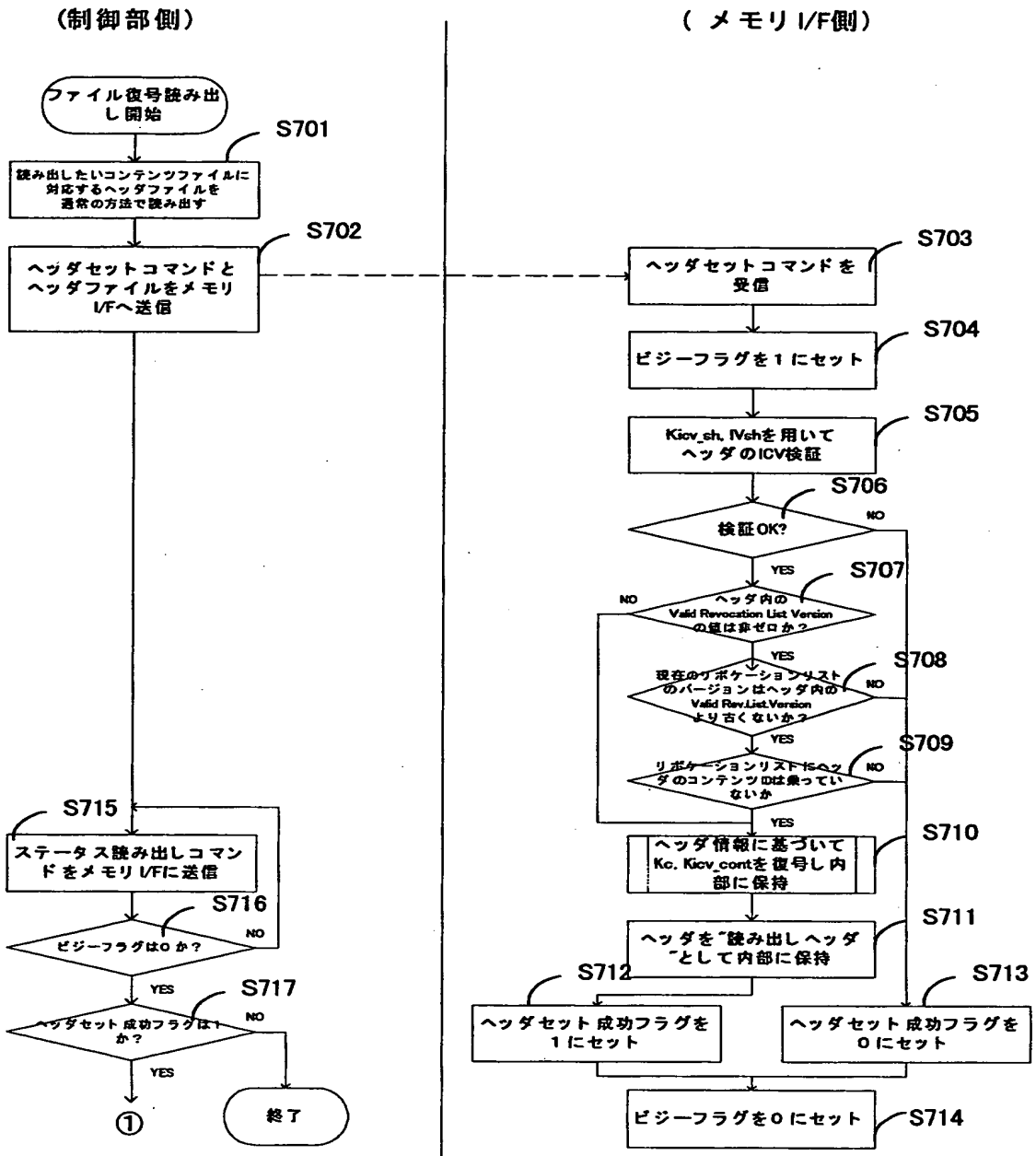
【図３３】


セキュリティヘッダ格納
セクタ対応
暗号化コンテンツキー
(メディア2の保存鍵
KstoでCBCモード暗号化)
E0
=Kc(0)Encrypted
E1
=Kc(1)Encrypted
E2
=Kc(2)Encrypted
EM-1
=Kc(M-1)Encrypted
DES
DES
DES
DES
Ksto
Ksto
Ksto
Ksto
初期値
(Initial Value:IV_Keys)
Kc(0)
Kc(1)
Kc(2)
Kc(M−1)
セクタ対応
コンテンツキー

【図３４】

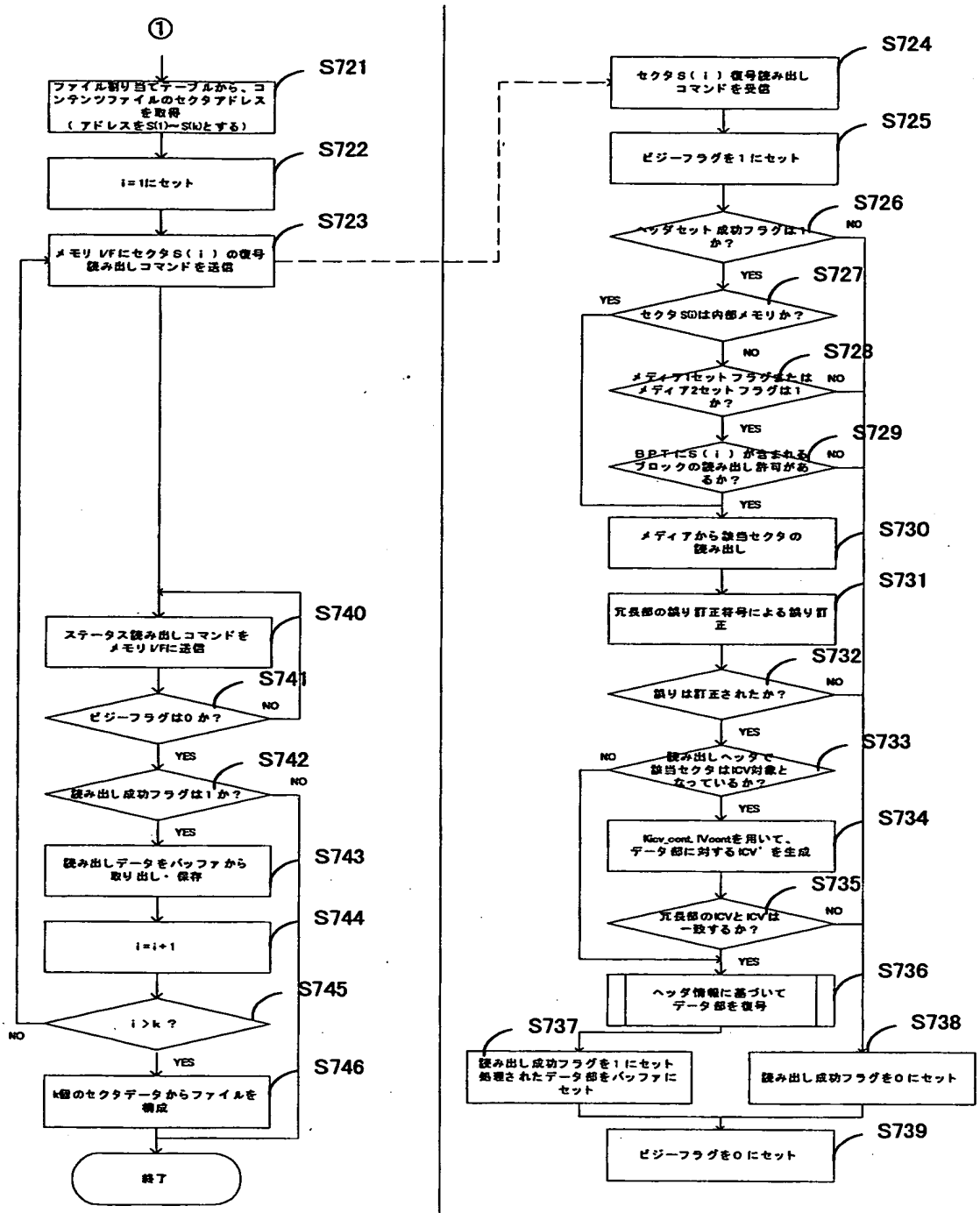

セキュリティヘッダ格納
セクタ対応
暗号化コンテンツキー
(メディア2の保存鍵
KstoでCBCモード暗号化)
E0
=Kc(0)Encrypted
E1
=Kc(1)Encrypted
E2
=Kc(2)Encrypted
EM-1
=Kc(M-1)Encrypted
DES復号
Ksto
初期値
(Initial Value:IV_Keys)
セクタ対応
コンテンツキー
Kc(0)
Kc(1)
Kc(2)
Kc(M−1)
DES暗号
Kses
セッションキーによる
CBCモード暗号化
セクタ対応コンテンツキー
SE0
=Kc(0)Encrypted
SE1
=Kc(1)Encrypted
SE2
=Kc(2)Encrypted
SEM-1
=Kc(M-1)Encrypted

【図３５】



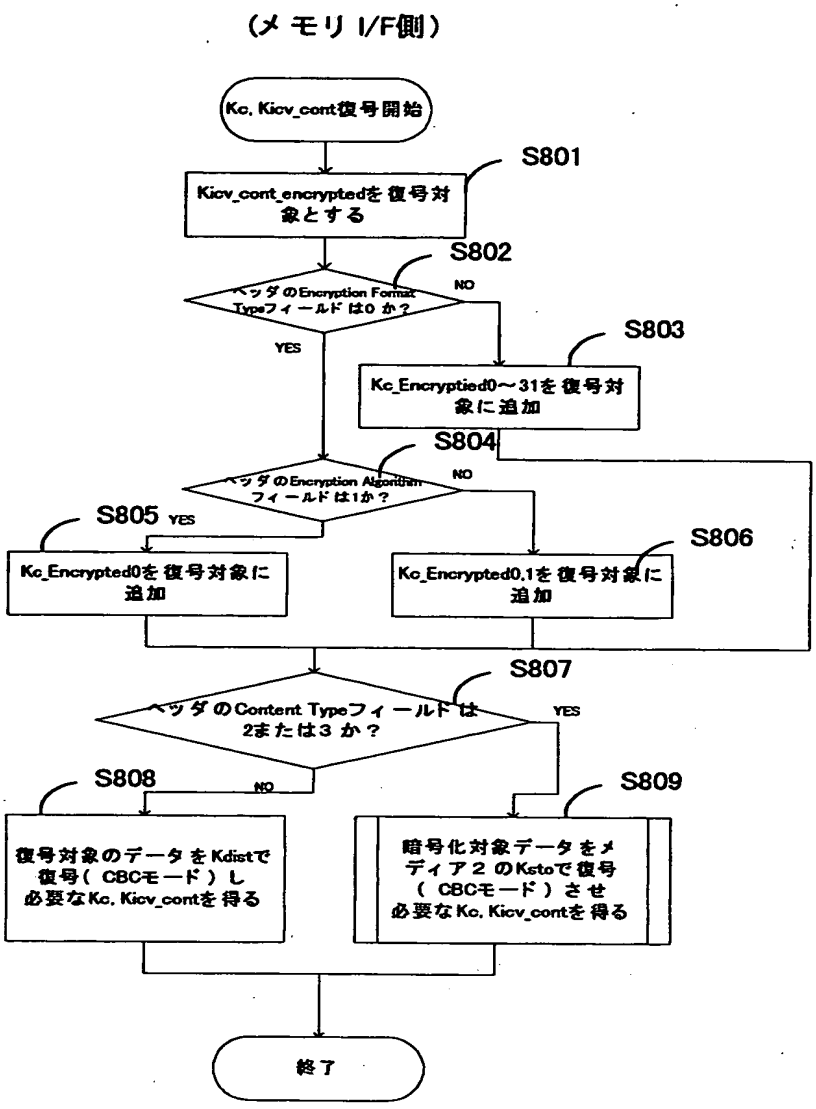
ファイルの復号読み出し処理

【図３６】



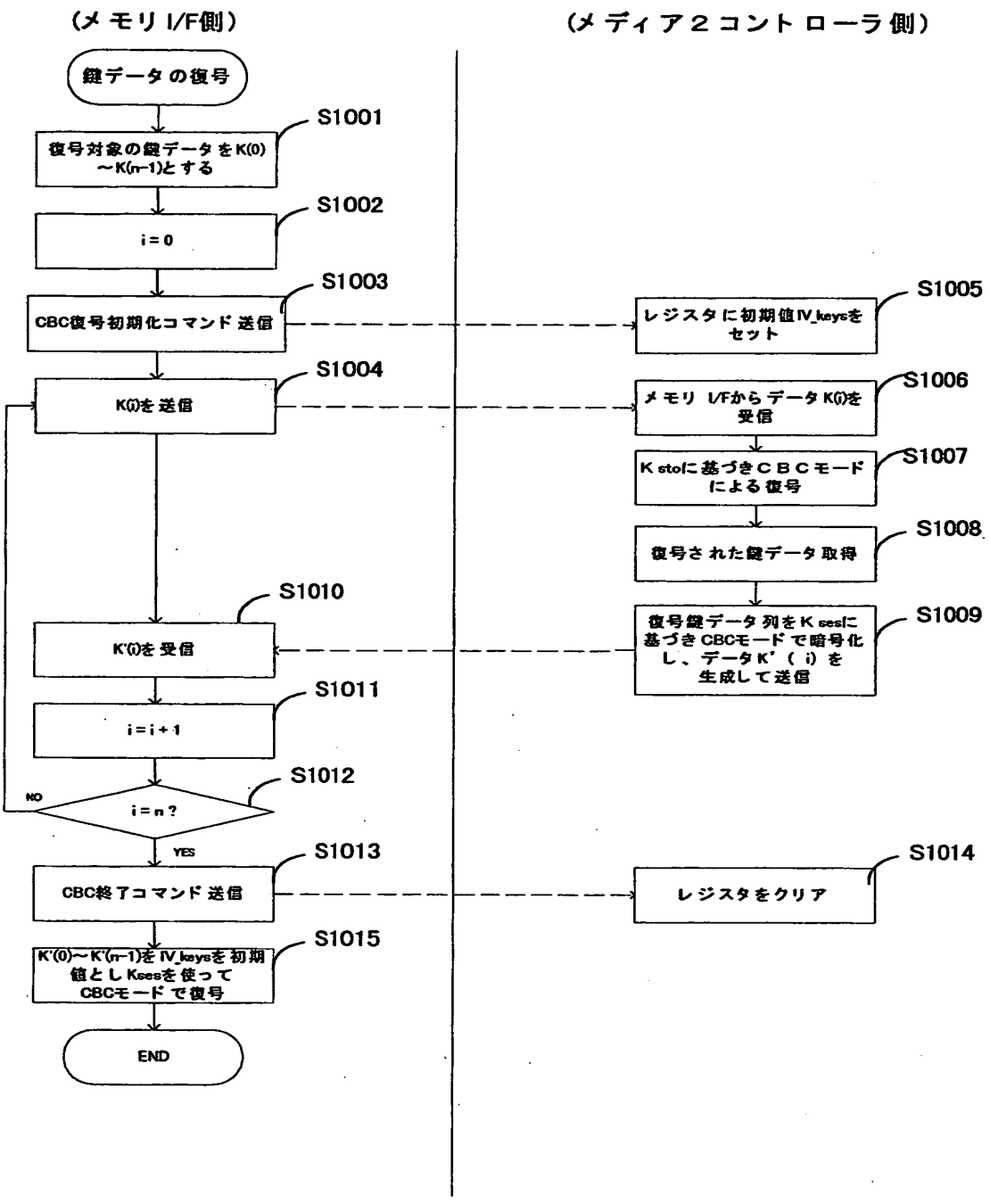
ファイルの復号読み出し処理

【図３７】



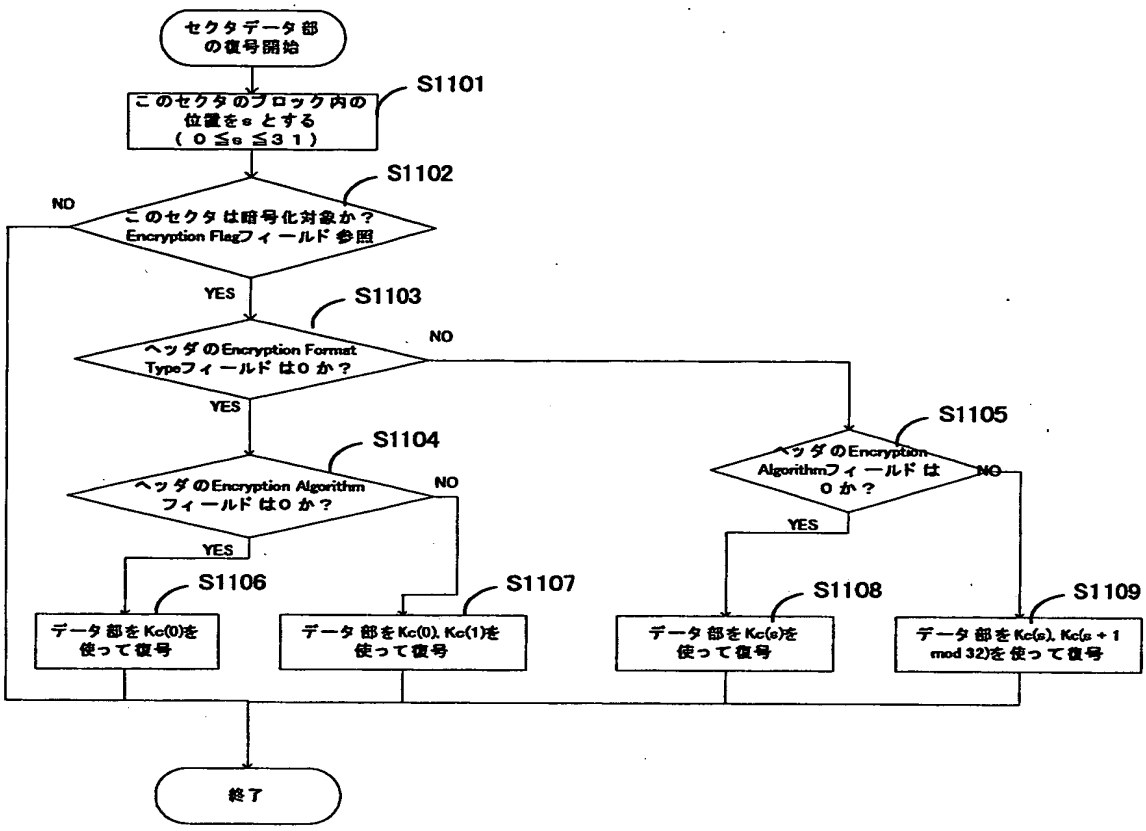
フロー 4-3 : Kc, Kicv_contの復号

【図３８】



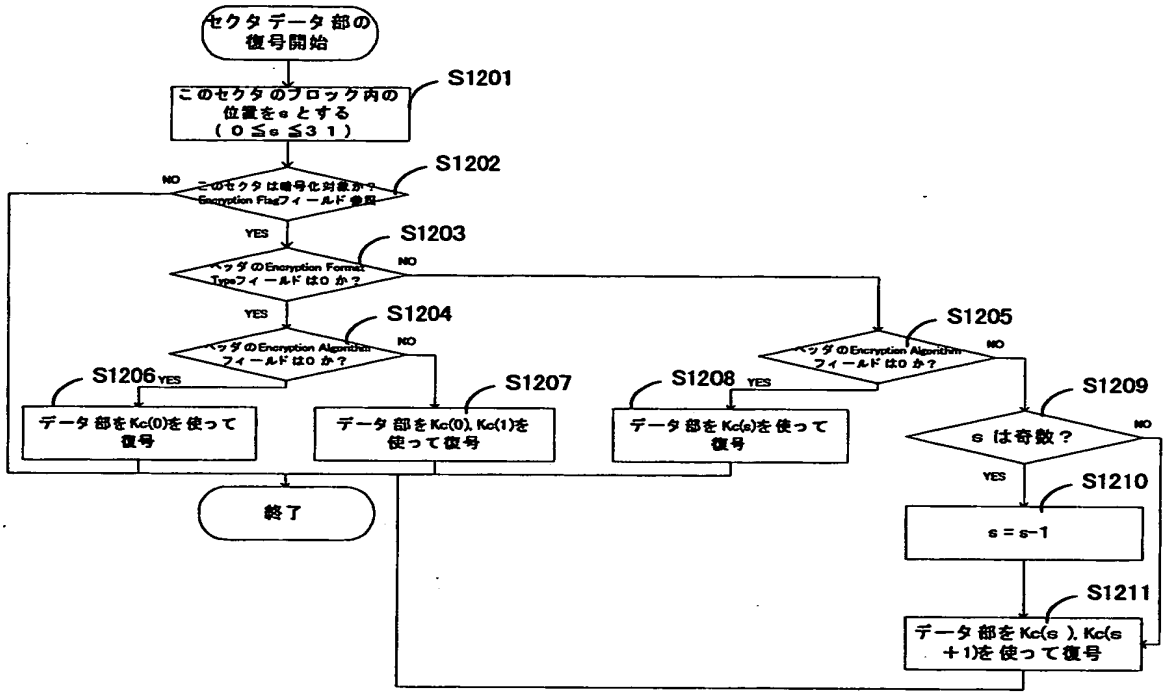
復号対象データをメディア２のKstoで復号

【図３９】



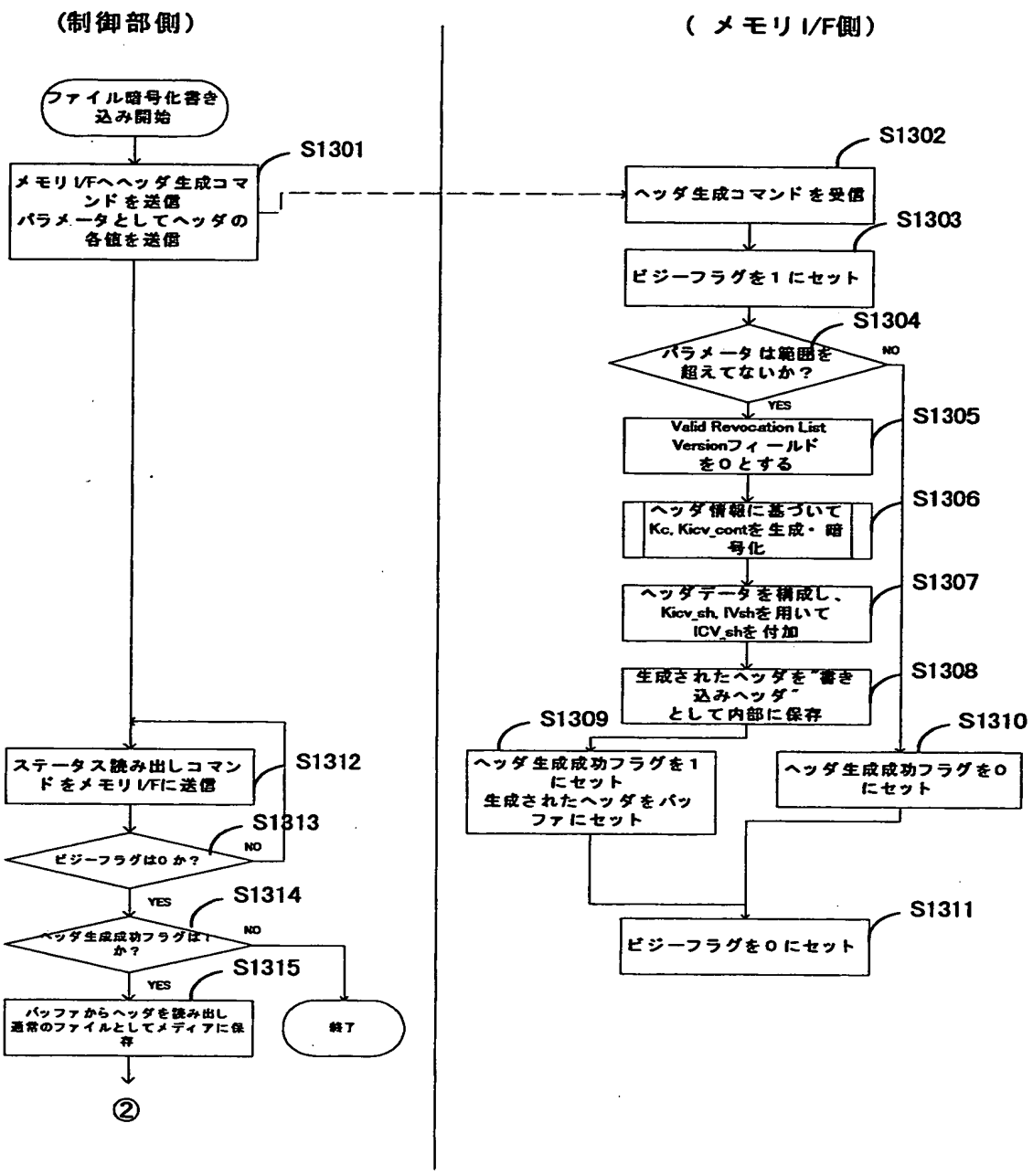
セクタデータ部の復号（ その１ ）

【図４０】



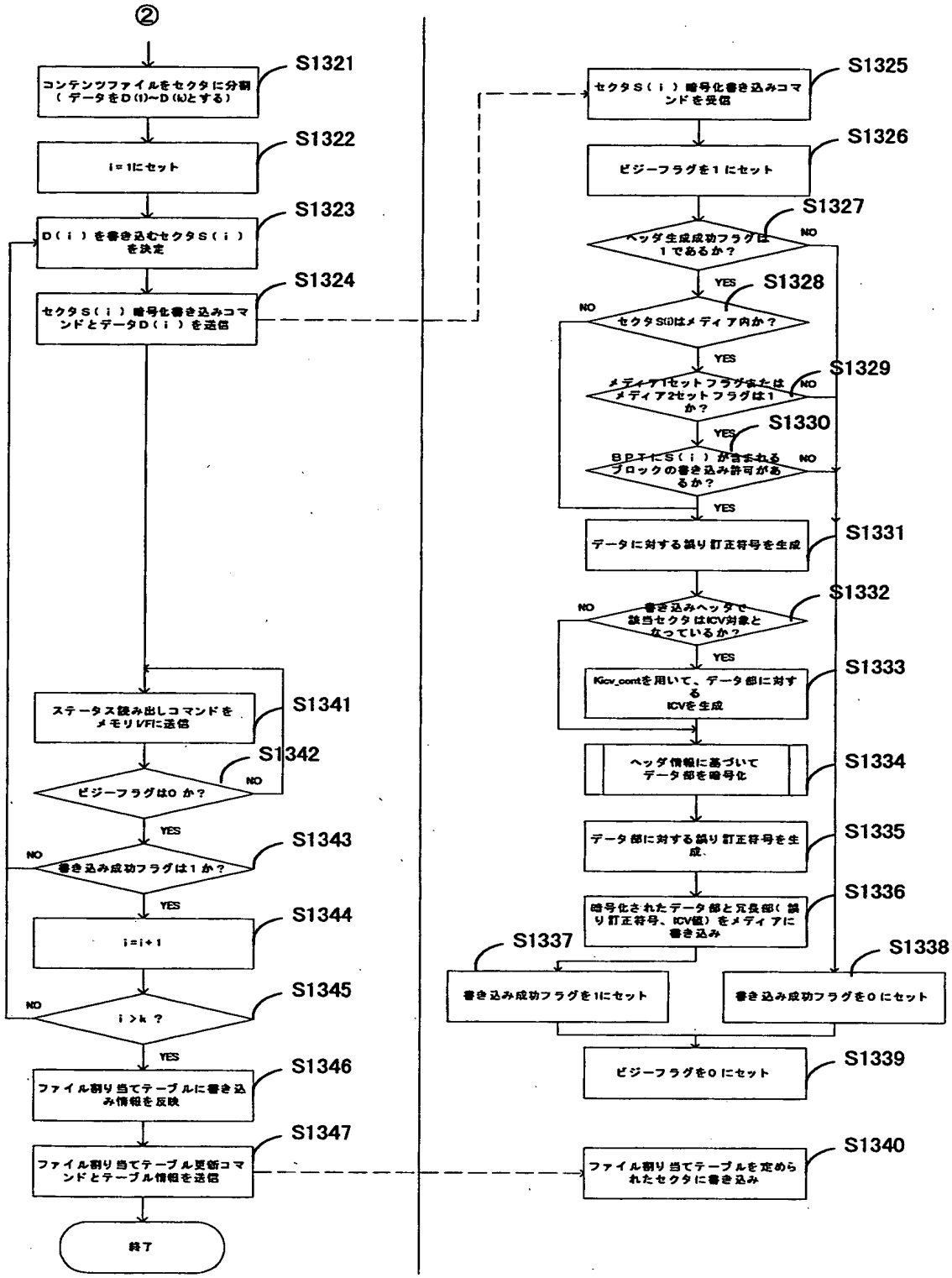
セクタデータ部の復号（その２）

【図４１】



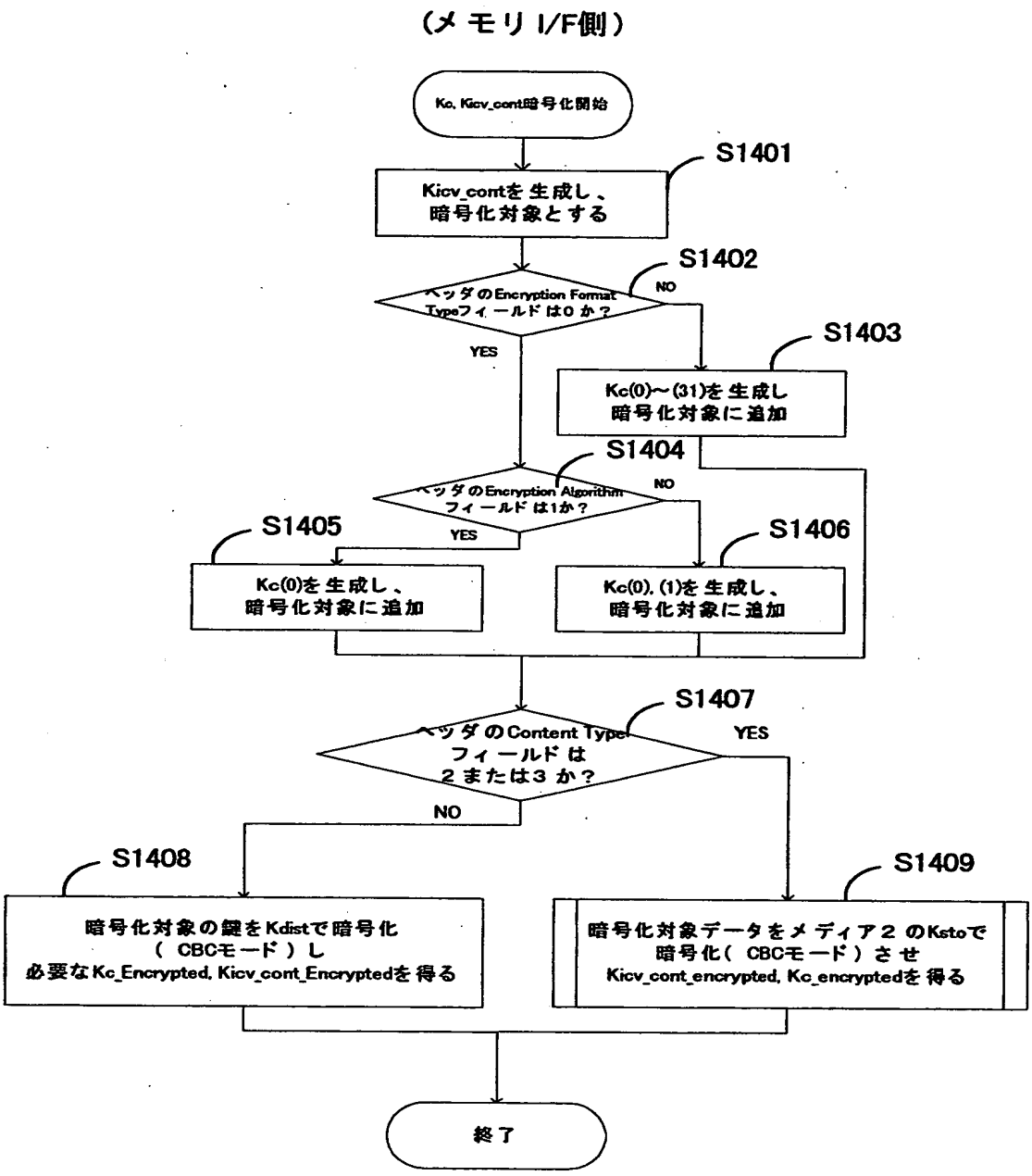
ファイルの暗号化書き込み処理

【図４２】



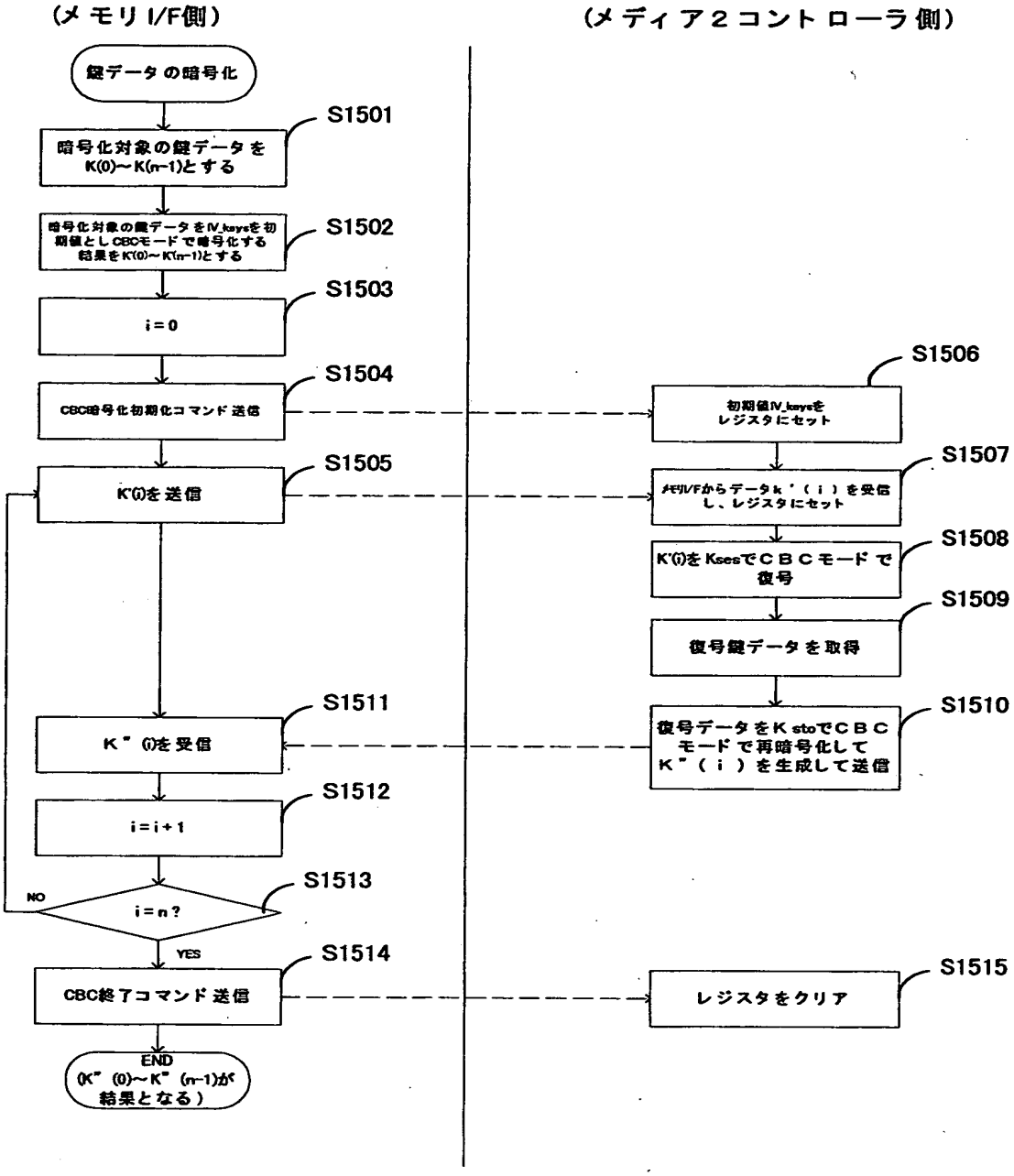
ファイルの暗号化書き込み処理

【図４３】



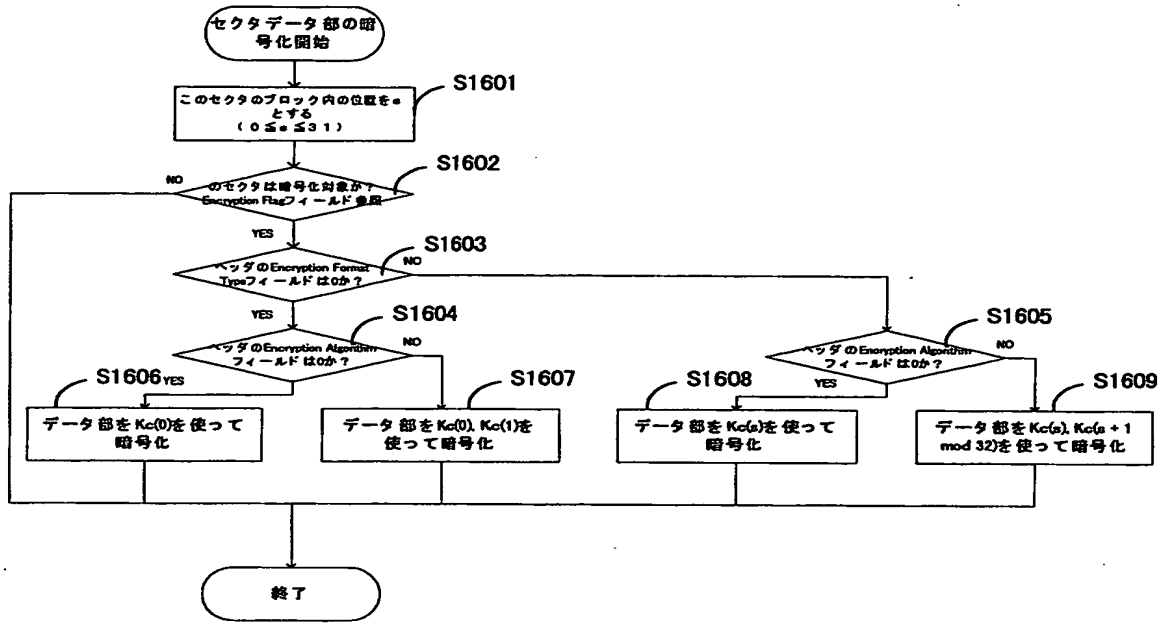
Kc, Kicv_contの暗号化

【図４４】



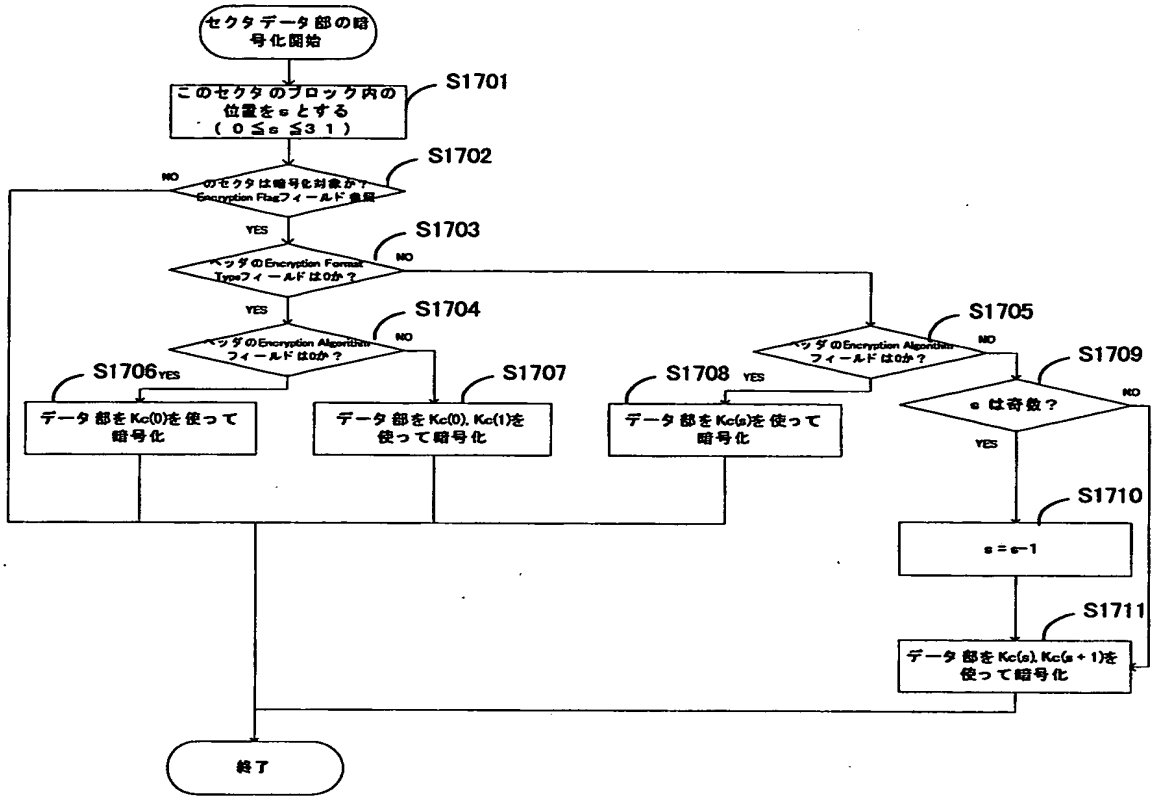
暗号化対象データをメディア２のKstoで暗号化

【図４５】



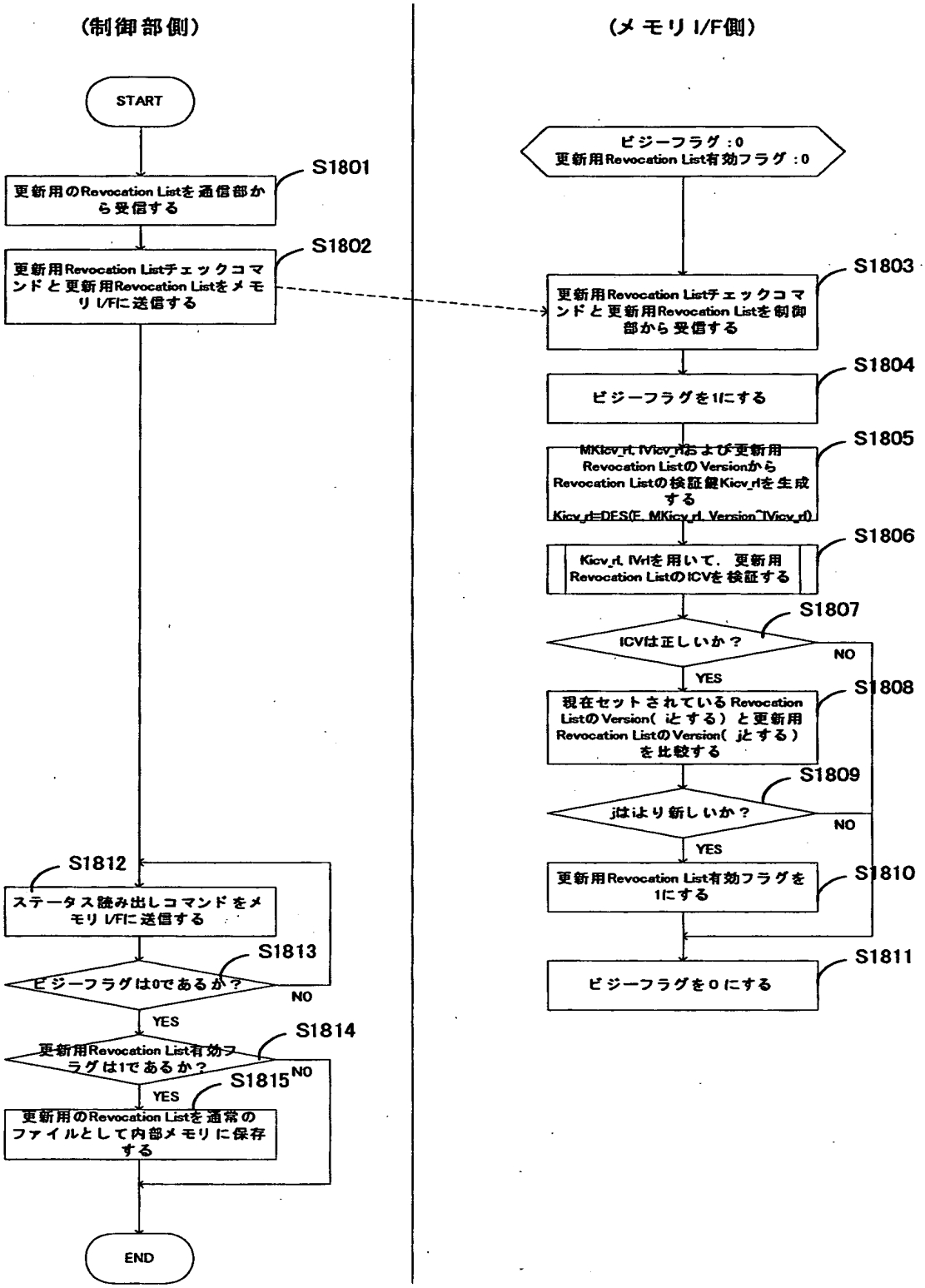
セクタデータ部の暗号化(その１)

【図４６】



セクタデータ部の暗号化（その２）

【図４７】



Revocation Listの更新

【書類名】　要約書

【要約】

【課題】　記憶手段に記憶するデータの暗号化コンテンツのセキュリティを高め、かつ鍵格納構成を改善した情報記録装置を提供する。

【解決手段】　コンテンツのセクタ対応の複数の暗号鍵について、データ記録再生装置とデータ記憶装置間においてＣＢＣモードでの暗号化処理を実行する。ＣＢＣモードで暗号化された結果のデータをコンテンツに対応するヘッダに格納する構成とした。ＣＢＣモードによる鍵の暗号化処理は、コンテンツを格納するメディアに固有の保存鍵を適用して行なう構成とし、例えばコンテンツ利用時には、相互認証の成立したメディアにおいて鍵データを復号することによってのみ鍵を復号して取得することが可能となり、セキュリティの高い鍵保管が実現される。

【選択図】　図３４

出　願　人　履　歴　情　報

識別番号　　〔０００００２１８５〕

１．変更年月日　　１９９０年　８月３０日
［変更理由］　新規登録
住　所　東京都品川区北品川６丁目７番３５号
氏　名　ソニー株式会社